National Library of Canada
Acquisitions and Bibliographic Services Branch
395 Wellington Street
Ottawa, Ontario
K1A 0N4

Bibliothèque nationale du Canada
Direction des acquisitions et des services bibliographiques
395, rue Wellington
Ottawa (Ontario)
K1A 0N4

Your file Votre référence

Our file Notre référence

NOTICE

The quality of this microform is heavily dependent upon the quality of the original thesis submitted for microfilming. Every effort has been made to ensure the highest quality of reproduction possible.

If pages are missing, contact the university which granted the degree.

Some pages may have indistinct print especially if the original pages were typed with a poor typewriter ribbon or if the university sent us an inferior photocopy.

Reproduction in full or in part of this microform is governed by the Canadian Copyright Act, R.S.C. 1970, c. C-30, and subsequent amendments.

AVIS

La qualité de cette microforme dépend grandement de la qualité de la thèse soumise au microfilmage. Nous avons tout fait pour assurer une qualité supérieure de reproduction.

S'il manque des pages, veuillez communiquer avec l'université qui a conféré le grade.

La qualité d'impression de certaines pages peut laisser à désirer, surtout si les pages originales ont été dactylographiées à l'aide d'un ruban usé ou si l'université nous a fait parvenir une photocopie de qualité inférieure.

La reproduction, même partielle, de cette microforme est soumise à la Loi canadienne sur le droit d'auteur, SRC 1970, c. C-30, et ses amendements subséquents.

Canada

UNIVERSITY OF ALBERTA

Three Dimensional Reconstruction of Trunk Surface Using Structured Light

BY

Richard Y. Liu

A thesis submitted to the Faculty of Graduate Studies and Research in partial fulfillment of the requirements for the degree of Masters of Science.

DEPARTMENT OF ELECTRICAL ENGINEERING

Edmonton, Alberta
Fall 1995

National Library of Canada
Acquisitions and Bibliographic Services Branch
395 Wellington Street
Ottawa, Ontario
K1A 0N4

Bibliothèque nationale du Canada
Direction des acquisitions et des services bibliographiques
395, rue Wellington
Ottawa (Ontario)
K1A 0N4

Your file *Votre référence*

Our file *Notre référence*

THE AUTHOR HAS GRANTED AN IRREVOCABLE NON-EXCLUSIVE LICENCE ALLOWING THE NATIONAL LIBRARY OF CANADA TO REPRODUCE, LOAN, DISTRIBUTE OR SELL COPIES OF HIS/HER THESIS BY ANY MEANS AND IN ANY FORM OR FORMAT, MAKING THIS THESIS AVAILABLE TO INTERESTED PERSONS.

L'AUTEUR A ACCORDE UNE LICENCE IRREVOCABLE ET NON EXCLUSIVE PERMETTANT A LA BIBLIOTHEQUE NATIONALE DU CANADA DE REPRODUIRE, PRETER, DISTRIBUER OU VENDRE DES COPIES DE SA THESE DE QUELQUE MANIERE ET SOUS QUELQUE FORME QUE CE SOIT POUR METTRE DES EXEMPLAIRES DE CETTE THESE A LA DISPOSITION DES PERSONNE INTERESSEES.

THE AUTHOR RETAINS OWNERSHIP OF THE COPYRIGHT IN HIS/HER THESIS. NEITHER THE THESIS NOR SUBSTANTIAL EXTRACTS FROM IT MAY BE PRINTED OR OTHERWISE REPRODUCED WITHOUT HIS/HER PERMISSION.

L'AUTEUR CONSERVE LA PROPRIETE DU DROIT D'AUTEUR QUI PROTEGE SA THESE. NI LA THESE NI DES EXTRAITS SUBSTANTIELS DE CELLE-CI NE DOIVENT ETRE IMPRIMES OU AUTREMENT REPRODUITS SANS SON AUTORISATION.

ISBN 0-612-06502-2

Canada

UNIVERSITY OF ALBERTA

RELEASE FORM

NAME OF AUTHOR: Richard Y. Liu

TITLE OF THESIS: Three Dimensional Reconstruction of Trunk Surface Using Structured Light

DEGREE: Masters of Science

YEAR THIS DEGREE GRANTED: 1995

Permission is hereby granted to the University of Alberta Library to reproduce single copies of this thesis and to lend or sell such copies for private, scholarly or scientific research purposes only.

The author reserves all other publication and other rights in association with the copyright in the thesis, and except as hereinbefore provided neither the thesis nor any substantial portion thereof may be printed or otherwise reproduced in any material form whatever without the author's prior written permission.

(Signed) . . . Richard . Liu . . . .

Richard Y. Liu
3439-78 Street
Edmonton, Alberta
Canada, T6k 0E1

Date: . . 25 . July . 1995

UNIVERSITY OF ALBERTA

FACULTY OF GRADUATE STUDIES AND RESEARCH

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies and Research for acceptance, a thesis entitled **Three Dimensional Reconstruction of Trunk Surface Using Structured Light** submitted by Richard Y. Liu in partial fulfillment of the requirements for the degree of Masters of Science.

N.G. Durdle (Co-Supervisor)

A.Peterson (Co-Supervisor)

J. Mowchenko

J. Raso

S. Bamforth

Date: 25 July 1995

To my parents and Meng, for their encouragement and support.

ABSTRACT

The study and monitoring of the three-dimensional trunk surfaces is important in the evaluation of scoliotic deformities. This thesis investigates and develops a method to acquire three-dimensional coordinate information of human trunk surfaces. Different three-dimensional data acquisition techniques are discussed in this thesis. Representative systems developed using stereo vision methods and structured light techniques are presented. An automatic system for recording the three-dimensional coordinates of a human trunk surface has been developed. The three-dimensional data acquisition process consists of image acquisition, pattern recognition, image rectification, and three-dimensional reconstruction. Image processing methods, such as region growing, edge detection, image thresholding, region thinning, line search, eight connectivity and line drawing are used to detect and extract the projected lines. Line correspondence between stereo image pairs and image rectification are used to match corresponding points between two images. The collinearity equations are used to recover the three-dimensional information. A system using structured light to reconstruct trunk surfaces is developed. The accuracy and reliability of the image algorithms are investigated. The system was tested using solid models. In addition, the system was used to image scoliosis patients to confirm the practicality of the method. The results show that it has the potential of becoming a valid clinical tool.

ACKNOWLEDGEMENTS

The author would like to express his deep gratitude to his supervisors Dr. N. Durdle and Dr. A. Peterson for their support and help during the course of this research work. Sincere thanks are also due to Mr. J. Raso for his participation in the supervisory committee, and to Mr. D. Hill for reviewing the manuscript and providing helpful suggestions.

The author is grateful to Dr. J. Mowchenko, and Dr. S. Bamforth, for their valuable suggestions in producing this thesis.

Thanks also go to Mr. K. denHartigh, for his technical assistance on IBM RISC System/6000 workstations.

The financial assistance provided by the Department of Rehabilitation Engineering at the Glenrose Rehabilitation Hospital during the summer and the Department of Electrical Engineering in form of Graduate Teaching Assistantship is greatly appreciated.

The author is happy to acknowledge his friends Mr. R.B. Penner, Mr. M. Polak, Mr. K. Bhalla and Mr. HeJim and others, for their help and assistance.

TABLE OF CONTENTS

1 INTRODUCTION 1

1.1 Purpose 1

1.2 Scoliosis 1

1.3 Scoliosis Measurement Methods 2

1.4 Motivation 5

1.5 Overview 6

2 3D DATA ACQUISITION SYSTEM REVIEWS 11

2.1 Scanning Method 12

2.1.1 Optical Time of Flight 12

2.1.2 ISIS Scanner 13

2.2 Stereo Vision Method 14

2.2.1 Feature Detection 15

2.2.2 Stereo Matching 16

2.2.3 Depth Calculation 21

2.3 Structured Light Method 23

2.3.1 Spot Projection 24

2.3.2 Light-Strip Method 25

2.3.3 Grid Pattern Approach 26

2.3.4 Single Camera System Depth Calculation Techniques 26

2.4 Camera Calibration Techniques 27

2.4.1 Standard Photogrammetric Methods 28
2.4.2 Single Camera and Single Projector Calibration Technique 30
2.5 Section Summary 31

3 SYSTEM DESIGN AND CALIBRATION **33**
3.1 Apparatus 33
3.1.1 Video System Design 33
3.1.2 Projector Slide Design 37
3.2 Video Camera Calibration 39
3.2.1 Camera Calibration Control Field 39
3.2.2 Calibration Procedure 43
3.3 Calibration Experimentation 43
3.3.1 Calibration Results 45

4 IMAGE FEATURE EXTRACTION **47**
4.1 Image Segmentation 47
4.1.1 Region Growing 48
4.1.2 Edge Detection 53
4.1.3 Segmentation of The Projected Line Image 55
4.2 Extraction of One-Pixel-Thick Lines 57
4.2.1 Image Thresholding 58
4.2.2 Line Extraction 59
4.3 Line Search 64
4.3.1 Three Stages of Line Search Procedure 64
4.3.2 Line Reconnecting Method 68
4.4 Line Sequence Analysis 71

5 THREE-DIMENSIONAL DATA ACQUISITION **73**

5.1 Line Correspondence Between Stereo Image Pair 73
5.2 Image Rectification 79
5.2.1 Background of Image Rectification 79
5.2.2 Implementation Of Image Rectification 81
5.3 Point Correspondence 84
5.4 Three-Dimensional Surface Reconstruction 84
5.4.1 Computation of 3D Data Using Collinearity Equations 85
5.4.2 Calculation of Initial Approximation of X, Y and Z 86
5.4.3 Application of Least Squares Solution 89

6 SYSTEM TESTING **92**
6.1 System Testing Using Object Models 92
6.2 Recovery of 3D Information from Scoliosis
Patient Trunk Surfaces 101

7 CONCLUSION **107**
7.1 Conclusion 107
7.2 Limitations 108
7.3 Future Work 110

Bibliography **112**

Appendices: **117**

1 Description of The 3D Reconstruction Package **118**
1.1 Implementation of Image processing techniques 119
1.2 Implementation of Three-Dimensional Reconstruction 125

LIST OF FIGURES

1.1 A person with scoliosis after surgical correction 3
1.2 Structure light system with two optical components 7
1.3 Regular spacing point pattern 8
1.4 Regular spacing point pattern projected on a body surface 9

2.1 Schematic diagram of optical time-of-flight 13
2.2 Correspondent problem between two images 17
2.3 The geometry of the epipolar lines. The plane defined by the two foci, F1 and F2, and a point P cuts the two image planes in the corresponding epipolar lines L1 and L2. 18
2.4 Point C violates the ordering constraint. 19

3.1 Structure light system with three optical components. 35
3.2 Parallel line pattern projected on a body surface. 36
3.3 A image of coded line pattern. 38
3.4 A control frame mounted on a object stand. 40
3.5 Image of the control frame. 41
3.6 Camera calibration procedures. 44

4.1 Object image with a spot lighting. 49
4.2 Resultant image after applying the region growing technique. 50
4.3 An example when region growing process fail. 51
4.4 Resultant image after pixel aggregation procedure. 52

4.5 Sobel masks: (a) 3 by 3 image region; (b) mask used to compute Gx; (c) mask used to compute Gy. 54
4.6 Enclosed boundary of the trunk. 56
4.7 Segmented projected line image. 57
4.8 Projected line image after local thresholding. 60
4.9 Neighbourhood arrangement used by the thinning algorithm. 61
4.10 Projected lines image after applying thinning algorithm. 63
4.11 An example of sequence of neighbouring broken lines. 66
4.12 Sequence of processing steps for the step pattern object. 68
4.13 The pixel grid for the midpoint line drawing algorithm. 70
4.14 Neighbourhood arrangement used by the eight-connectivity. 72

5.1 Sequence of processing steps for the pyramid shaped object. 75
5.2 Line correspondence between the stereo pair. 76
5.3 Occlusion in the stereo image pair. 78
5.4 Parallel epipolar lines in both images. 79
5.5 Geometry of image rectification. 80
5.6 Geometry of 3D reconstruction. 87
5.7 Reconstructed model of a human trunk. 91

6.1 Sequence of processing steps for the cylindrical object. 94
6.2 Y and Z coordinate information of the cylindrical object. 95
6.3 Sequence of processing steps for the step pattern object. 97
6.4 X and Y coordinate information of the step pattern object. 98
6.5 Sequence of processing steps for the triangular shaped object. 100
6.6 Y and Z coordinate information of the triangular shaped object. . . . 101
6.7 Sequence of processing steps for the scoliosis patient. 103
6.8 Sequence of processing steps for the scoliosis patient. 105
6.9 Sequence of processing steps for the scoliosis patient. 106

LIST OF TABLES

3.1 Control point XYZ coordinates. 42

3.2 Four sets of camera calibration results. 45

6.1 Accuracy of the eight trials of the reconstructed cylindrical object. . . 93

6.2 Accuracy of the angles of the reconstructed triangular shaped object. 99

Chapter 1

INTRODUCTION

1.1 Purpose

The objective of this research is to investigate and develop a method to acquire three-dimensional coordinate information from the trunk surfaces of a scoliosis patient, and to provide clinicians with a tool to assist in the evaluation of scoliotic deformities. This thesis presents background information on three-dimensional data acquisition techniques and describes the video system in place at the Glenrose Rehabilitation Hospital. In addition, an implementation of a depth information acquisition process is described. Accuracy of the acquired three dimensional data is studied, limitations of the system are also presented.

1.2 Scoliosis

Scoliosis causes lateral bending and axial rotation of the spine. Scoliosis often causes asymmetries of the trunk, prominence of one of the shoulder blades, and a subtle twisting of the trunk(Figure 1.1). The spinal rotation causes the rib-cage to distort and produces a hump on one side of the trunk. Scoliosis can occur in both genders at any age; however, it mostly occurs in adolescence females. It affects about 0.3% of the population [53]. In spite of numerous investigations into possible causes, little

is known about the causes of scoliosis, but, it is generally thought that it results from more than one factor [10]. There are several physiological anomalies in bony structures, neuromuscular tissues, hormones, and collagen associated with scoliosis, but, whether they are the cause or the effect has not been determined. Since so little is known about its cause, there is no preventative treatment for scoliosis. Once scoliosis is diagnosed, there are corrective treatments , such as bracing, casting, and surgery. Casts and braces apply a de-rotational force rather than a distraction force. Surgery involves attaching rods to vertebrae to apply an axial distraction force and de-rotational forces.

1.3 Scoliosis Measurement Methods

Monitoring and measuring the severity of spinal deformity is necessary when assessing the effect of treatment and for historical tracking. The most widely used technique to assess the severity of scoliosis is standard back-to-front spine radiography. Radiological assessment of curvature in scoliosis usually includes measurement of the Cobb angle [9] between vertebrae on spinal radiographs. The clinical definition of scoliosis is a curve which has a Cobb angle exceeding ten degrees and is accompanied with vertebral rotation. Radiographic methods give a picture of the deformed spine, and can establish the location of the spinal curves. To extract the three dimensional information of the spine, at least two radiographs taken from different angles are required. Radiography can not be used to assess surface shapes or for cosmetic treatment, because only bony elements are seen well on the radiograph [54]. The cosmetic deformity is only weakly correlated with the underlying spinal misalignment and these need to be studied with a different modality. To minimize radiation and financial cost, a single back-to-front radiograph is often used to assess scoliosis. Clinicians can not accurately estimate the 3-dimensional deformity of the spine from one radiograph. In addition, radiography is an invasive technique. Ongoing irradiation of

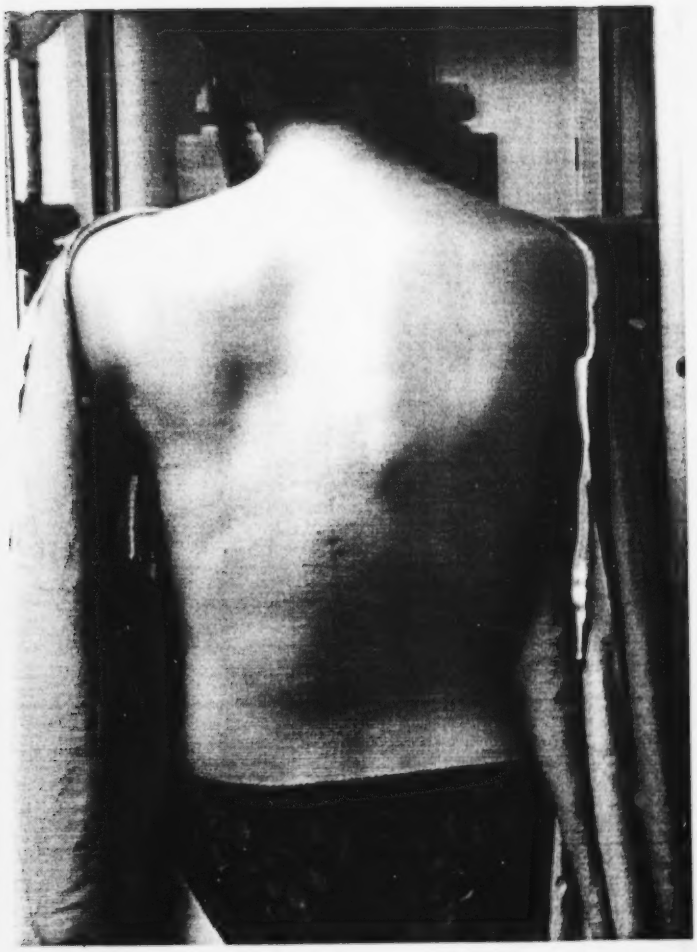

Figure 1.1: A person with scoliosis after surgical correction

the patient may be undesirable, particularly in a growing child if frequent monitoring of deformity progression is required.

A complimentary method for analysing the development of scoliosis is to measure the three-dimensional surface deformity of the trunk. Measurement of surface shape does not involve radiological techniques, and is non-invasive. There are many well established techniques for measuring three dimensional surface shapes. These include stereo photogrammetry, moire topography and structured light. With moire topography, the shadow of a grid is projected onto the patient. When observed through a second grid, interference fringes appear on the body surface. These fringes, moire fringes, correspond to the depth contours on the body surface. The surface shape of the patient can be recovered from the position of each fringe and the geometry of the projection and recording apparatus. However, this method is subject to large errors due to changes in the fringe pattern resulting from patient movement [49]. In addition, the moire topographic method requires multiple observations to establish the direction of depth change. This causes difficulty in achieving a short observation time, which is necessary when taking images of a patient. Patient movement due to breathing and posture sway may mask trunk shape changes. Moreover, the fringes produced by moire techniques are difficult to analyse by computer [52].

Another method for measuring three dimensional surface shape is stereo photogrammetry, [12],[39] and [29]. This method involves image feature extraction and determination of corresponding features in two camera images. This technique is non-invasive and allows free monitoring of the patient. Stereo photogrammetry can produce accurate and reliable three-dimensional information; however, it has not been considered for most medical applications. This is because the technique depends heavily on ambient light, a change in lighting conditions causing change of pixel intensity will introduce inaccuracies. Moreover, it requires complex feature extraction

and matching algorithms for automation, especially, when the scene is a smooth and featureless body surface.

Another optical method, commonly used in biomedical applications, is structured light, [14], [35], and [30]. It usually involves projectors and video cameras. In this method, a pattern of lines or dots is projected onto the body surface and an image is taken via the video cameras. After the images have been obtained, three dimensional coordinate information can be calculated by using the similar techniques to those used in stereo photogrammetry. The advantage of this technique over stereo photogrammetry is that it uses a projected light pattern adding specific features such as dots or lines to the featureless body surface. This reduces the complexity of the feature extraction and matching algorithms [21]. This method can provide an accurate description of the surface asymmetry of the trunk and can produce a visible three-dimensional surface output for the clinicians during scoliosis assessment. Moreover, it is safe and inexpensive compared to other measuring techniques. The projected light source is hazard free for both operator and patient.

1.4 Motivation

The Glenrose Rehabilitation Hospital is involved in the assessment of scoliosis in patients. The existing surface topography measurement system at the Glenrose Rehabilitation Hospital [31], shown in figure 1.2, consists of a video camera(JVC TK-S310U), a projector with regular dot patterns on a standard 35mm slide and a Macintosh II computer equipped with hardware capable of capturing a single frame in 1/30 of a second. The camera with a 16mm f/8 lens is placed below the projector. The system setup is based on a rasterstereography method described by Jeremy Pearson [45]. In the measurement system, a regular dot pattern(Figure 1.3) is projected onto the body surface. The pattern formed on the surface(Figure 1.4) of the body is acquired by

the video camera and transferred to the memory of the Macintosh II computer using the frame grabber. After manually digitizing and correlating points between object image and reference image, the three-dimensional shape of the object is obtained through a triangulation method. This method is non-contact and non-invasive and low cost. However, because an automatic image process matching algorithm is not implemented, image segmentation and the matching of corresponding points between two images are time consuming, and very labour intensive. In addition, because a manual technique is used to record clinical data, the existing measurement system requires a skillful person to operate it; hence, it has limited usefulness in scoliosis studies. To make the three-dimensional acquisition system more clinically useful, it is necessary to find a way to improve it, so that, the system can measure the surface coordinates of the human trunk with reasonable precision, perform image analysis and coordinate computation without human intervention, and record image data in a short time. Typical coordinates obtained by a system need to be accurate to about 2-3 mm [37], with a recording time for the system of 30-50 milliseconds.

At present, a graphical display for computer reconstructed 3D models is available at the Gelnrose Rehabilitation Hospital. However, an automatic system for producing three-dimensional information is not yet fully implemented. It is necessary to design and develop a new fully computerized three-dimensional data acquisition system so that clinicians can use the system as a tool to assist in the evaluation of scoliotic deformities.

1.5 Overview

Different three-dimensional data acquisition techniques were investigated and compared. The structured light technique was selected for recovery of three-dimensional data from human trunk surfaces. Structured light is the most common technique used

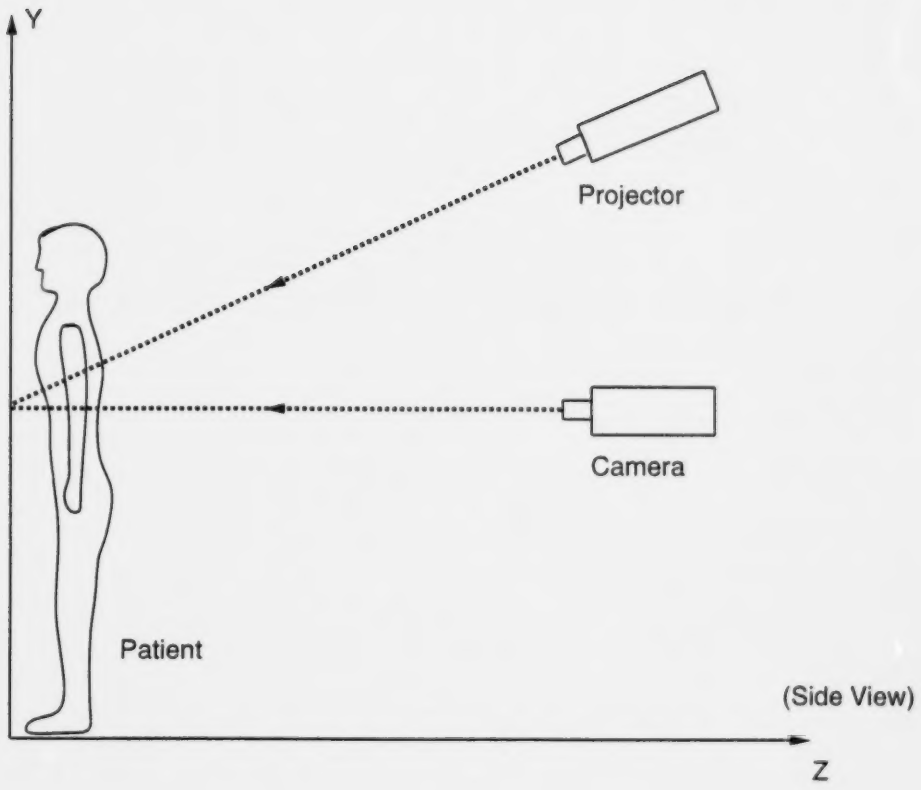

Figure 1.2: Structure light system with two optical components

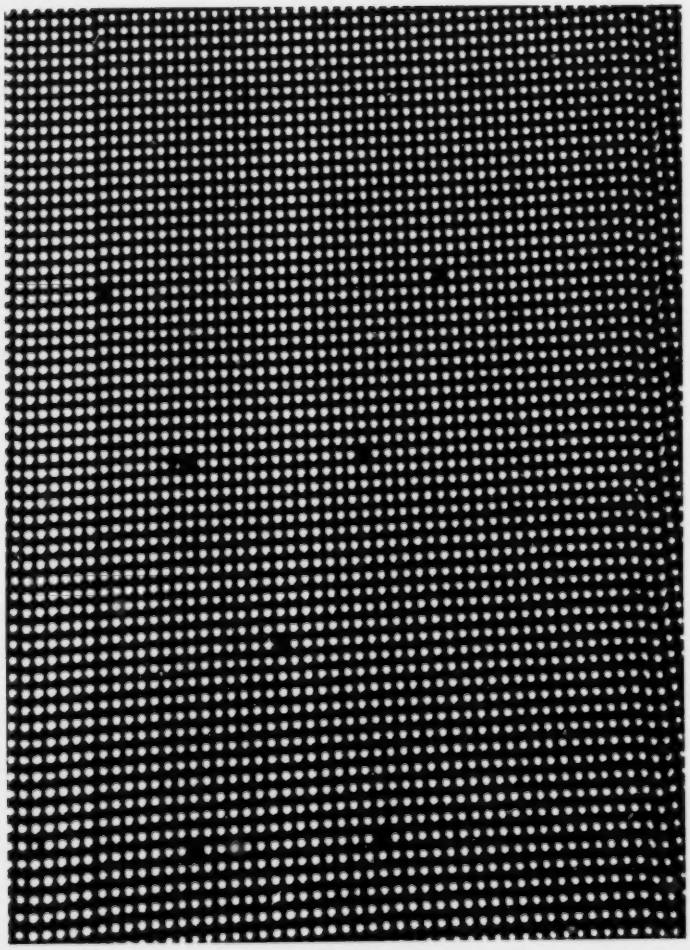

Figure 1.3: Regular spacing point pattern

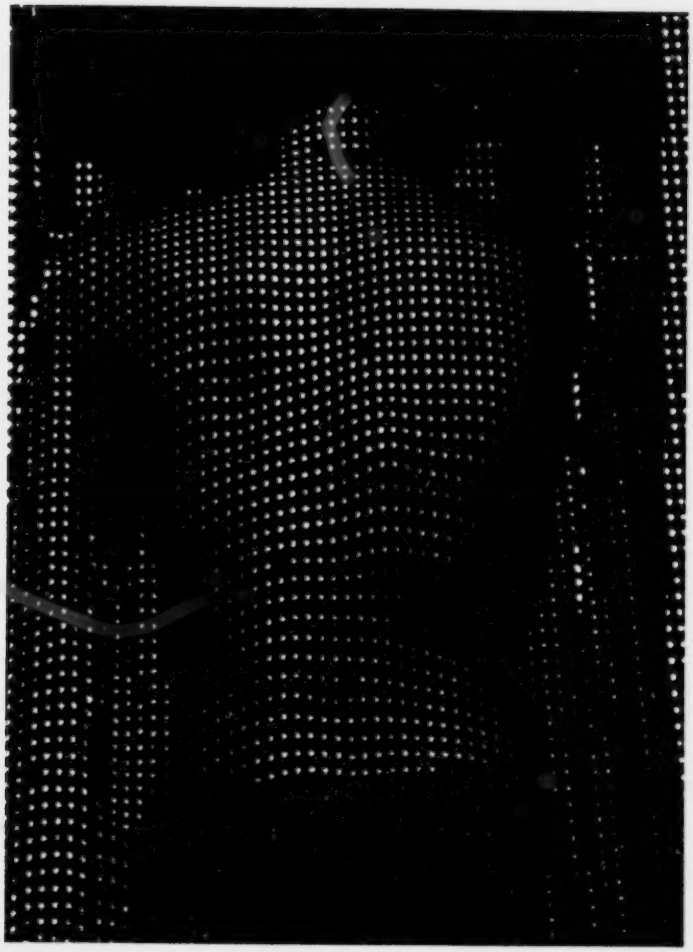

Figure 1.4: Regular spacing point pattern projected on a body surface

in machine vision. Many structured light systems have been developed. However, unlike many structured light systems used before, the system proposed by this research is based on stereography principles and requires only standard camera calibration and reconstruction procedures. The objective of this research is to develop an automatic system to record the three-dimensional coordinates of a human trunk surface.

Several different three-dimensional data acquisition techniques are discussed in Chapter 2 and representative systems developed using stereo vision methods and structured light techniques are presented. The review suggests that further investigation into development of an automatic system for recording the three-dimensional coordinates of a human trunk is required. An image acquisition system based on stereography principles requiring only standard camera calibration and reconstruction procedures is described in chapter 3. The description of the video camera system, the camera calibration procedure and the calibration results are also presented in chapter 3. Chapter 4 describes image processing methods employed to define one-pixel-thick lines for the projected line image. The methods include region growing, edge detection, image threshold, region thinning technique, line linking procedure, eight connectivity technique and line drawing algorithm. Chapter 5 presents line correspondence between the stereo image pair and image rectification technique. In addition, point correlation between two images and the method employed in reconstruction of three-dimensional information are also described in this chapter. Chapter 6 presents test results from the system. Solid models and scoliosis patients are used to test the application of the system. The results demonstrate the feasibility of the system for three-dimensional information recovery of the trunk surface. A conclusion, discussion of the limitation of the system, and recommended future work is presented in chapter 7. Description of the software implemented for the system is given in Appendix. Source codes are available from Dr. Nelson G. Durdle in the Department of Electrical Engineering at the University of Alberta.

Chapter 2

3D DATA ACQUISITION SYSTEM REVIEWS

There are several competing technologies for determining three-dimensional coordinates, but none of them satisfy all the requirements for all applications under all working environments. Therefore, alternative techniques to accurately determine the three-dimensional information of the human trunk require investigation. The technique to be used must satisfy the following constraints: 1) It must be able to measure surface coordinates of the human trunk with little or no human intervention. 2) Image analysis and corresponding feature matching between two images must produce reliable results, coordinate accuracy within 3 millimetres. 3) Image capture must be fast to minimize errors due to movement.

Three different optical techniques exist for finding three-dimensional information: scanning methods, passive methods(stereo vision), and active methods(structured light). Scanning methods involve the use of a laser, a light source, a scanning mechanism and a receiver. Passive methods involve a matching process between two images. Active methods project a specially designed light pattern and observe the projected images by one or more video cameras.

2.1 Scanning Method

Scanning methods usually involve a laser light source, a scanning mechanism and a receiver. This method is simple and efficient. It does not require matching corresponding points between two images; thus, it does not require adding features to featureless objects. Also it has the advantage of working under any lighting condition and has a large depth of field. However, because multiple images are required, the scanning systems have long recording times introducing the possibility of motion artifacts. There are many scanning devices that have been developed to reconstruct the three-dimensional information. Optical time of flight and Integrated Shape Imaging System (ISIS) are two representative approaches.

2.1.1 Optical Time of Flight

This approach typically involves use of a transmitter that emits a light bean, a scanning mechanism and a receiver that detects the reflected light from the target. The three-dimensional information is obtained from the time needed for the light to travel from the transmitter to the target and back to the receiver [2]. There are two methods used for determining the time of flight. In the first, a pulse laser is used. Time elapsed from emission to reception is measured as a function of depth information. This method [49] requires accurate measurement of time and generates inaccurate data if the pulse is deformed during reflection.

The second method is to modulate a laser beam in amplitude and measure the phase shift of reflected light. A typical system [49] is illustrated in figure 2.1. In this system, an instrument emits a laser beam that is amplitude-modulated with a sinusoidal wave form. The beam is deflected by the mirror in the scanning unit. When the beam strikes an object, the light is reflected back to the scanning unit, and deflected onto the receiver. The phase of the signal, relative to the reference phase,

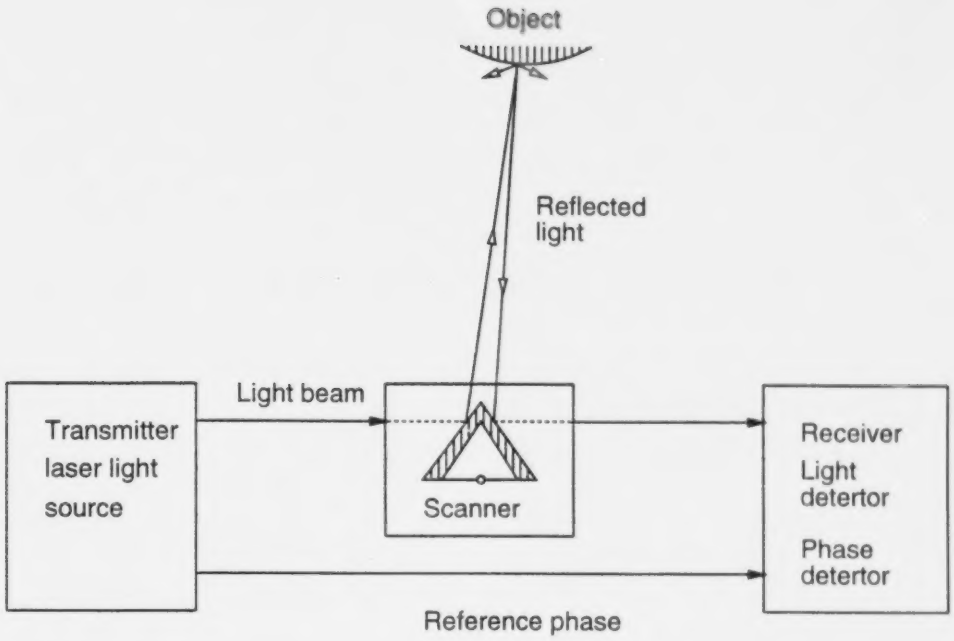

Figure 2.1: Schematic diagram of optical time-of-flight

is proportional to the time of flight. Thus the instrument can calculate the depth information. The problem is that the system requires a two-dimensional scanning mechanism and a complicated design.

2.1.2 ISIS Scanner

The Integrated Shape Imaging System (ISIS) was developed by Turner-Smith [52]. It was a commercially available device and was used primarily in the mid to late 1980's to measure trunk shape on children with scoliosis. The system consisted of a 35 mm projector and a computer-linked video camera mounted together in a box. The box could swing vertically about a horizontal axis. The projector produced a horizontal plane of light on a object surface. The video camera captured the image at different angles. Thus, the two-dimensional coordinates of a line was recorded directly by a computer. Depth information of the line was calculated from the known coordinates

of the camera and the geometry of the system. By swinging the camera/projector unit about its axis, a complete record of the three-dimensional coordinate information of the object surface was computed. This system did all the image processing and matching and produced three-dimensional data. The resolution of the data was about $\pm 1.5mm$, and the accuracy was approximately $\pm 3mm$ over a volume 400mm x 500mm x 300mm. However, the camera and the projector of this system required a special and complicated calibration scheme, and the accuracy of the depth information was heavily dependent on results of system calibration. In addition, this method required a step motor to control the swing of the camera/projector unit; thus, complicated system design was required. The major drawback of ISIS was the excessive time required to capture a full trunk. Posture changes due to sway and breathing introduced unacceptable levels of artifacts [23].

2.2 Stereo Vision Method

A stereo vision approach can be used to determine three-dimensional coordinates. It involves the use of two or more cameras. Correlating features between the two camera images is the critical stage in this method. Once the correlation problem is solved, three-dimensional coordinate information can be calculated based on the Direct Linear Transformation [1] or the collinearity equations [56]. The stereo vision method has the advantage of being less restricted for trunk measurement compared to the scanning method. It can be used to construct three-dimensional coordinates for general scenes. To obtain three-dimensional coordinate information, there are usually three steps required: 1) detecting features in each of the two camera images, 2) matching features between two images, and 3) computing depth information.

2.2.1 Feature Detection

Extracting features from the images is the first step in applying stereo vision techniques. It is usually not a good practice to try to find corresponding points for all pixels in the image [55]. In general, a point in one image may correspond to many points in the other image. Thus, feature extraction is an important step in the matching process.

Feature extraction involves determining a set of points from real-world images. This set of points is used to establish correspondence between two images. Often, the set of points is extracted from the features such as the zero-crossings of the Laplacian, corners or edges. Many image processing techniques are available to extract these features, eg. et al. [5], [20] and [13].

Marr and Poggio [41] proposed using zero-crossing method for feature extraction. Instead of finding the maxima of a intensity, this method found zero-crossings of the differentiated intensity in the image. Marr and Poggio suggested that the best operation was to apply the Gaussian operator to the original image for smoothing, and then apply the Laplacian operator. Later, Grimson [27] implemented his scheme by using a Laplacian of a Gaussian for detecting features. Zero-crossing of a image may correspond to edges, thus, a small fluctuation in the image may also produce zero-crossings. To avoid the effect of noise, in Grimson's system, zero-crossings that were produced by small light intensity changes were neglected. The way they achieved this was to examine the difference of intensity value across a zero-crossing.

Feature points also can be extracted on the basis of the gray level variation in their neighbouring regions. A point is regarded as a feature point if the gray level changes considerably in the neighbouring region. A typical way to find feature points is to apply operators, such as gradient operators, that are sensitive to gray level variance.

Feature points can then be obtained by thresholding the output value of the gradient operator [40]. This method works well for simple images, but it has difficulty with complex images.

2.2.2 Stereo Matching

While all three procedures are important to the process of three-dimensional data acquisition, stereo matching has usually been considered the critical one. In general, given two sets of points or segments in images 1 and 2 respectively, as shown in figure 2.2, the question is, which points on image 2 correspond to points B, F, and G on image 1? This is considered the matching problem. There are some constraints that can be applied to simplify the matching problem [17]. These constraints arise from the geometry of the stereo vision system:

1. **Uniqueness:** A point in one image matches only one point in the other image. The exception is when an occlusion occurs in one image but not the other. In this case, a point in one image is not matched with any point in the other image. This is also refereed to as the missing point problem.

2. **Continuity:** The assumption is that objects are generally smooth. Although discontinuities occur where depth changes sharply, these regions occupy much less of the image area than do smooth regions. Stereo disparity generally varies smoothly throughout the image.

3. **Contrast sign:** Stereo matching must take place between points which are either both bright or both dark; a dark point can not match a bright one.

4. **Ordering constraint along epipolar lines:** As shown in figure 2.3, given a point P in space, there exists a plane that is defined by P and the two camera foci. Epipolar lines are formed by the intersection of this plane with the two image planes. As a result, corresponding points between two images are located on a pair of epipolar lines. This implies that the matching problem can be reduced to a one-dimensional

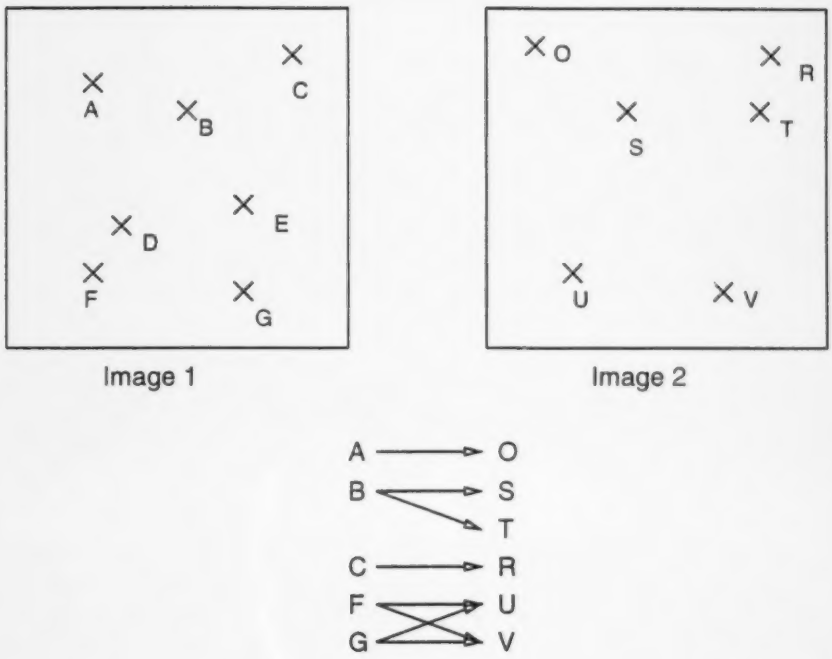

Figure 2.2: Correspondent problem between two images

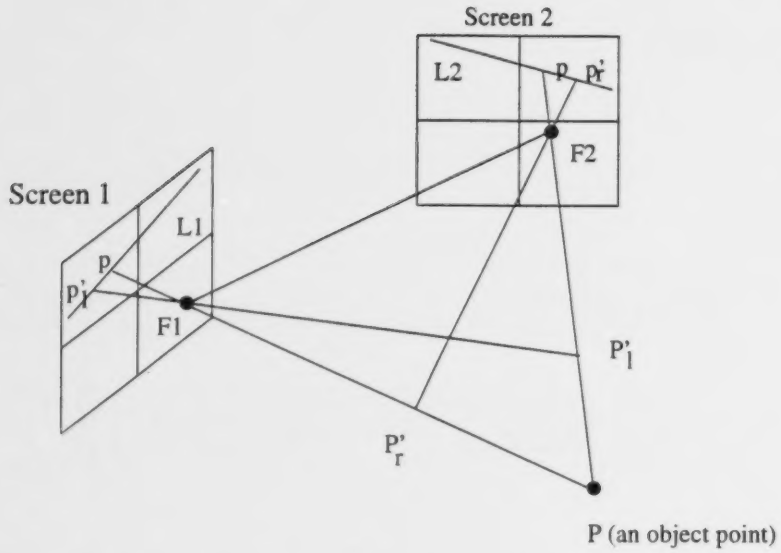

Figure 2.3: The geometry of the epipolar lines. The plane defined by the two foci, F1 and F2, and a point P cuts the two image planes in the corresponding epipolar lines L1 and L2.

search if the epipolar lines are known [40].

The ordering of edges or other features is usually the same along the epipolar lines. This means, as shown in figure 2.4, if feature A is on the right side of feature B in one image, then this spatial relationship is maintained in the other image as long as features A and B are not in the "forbidden zone". The forbidden zone [51] is defined as a set of points in space that would have images violating the ordering constraint.

Figure 2.4 shows an example with feature C in the "forbidden zone". The spatial relationship of features B and C in images 1 and 2 violates the ordering constraint. Feature B is on the right side of feature C in image 1, but feature B is on the left side of feature C in image 2. The ordering of the features B and C is not the same along the epipolar line. The ordering constraint along the epipolar line is violated. The epipolar line constraint can not be used to solve the correspondent problem in

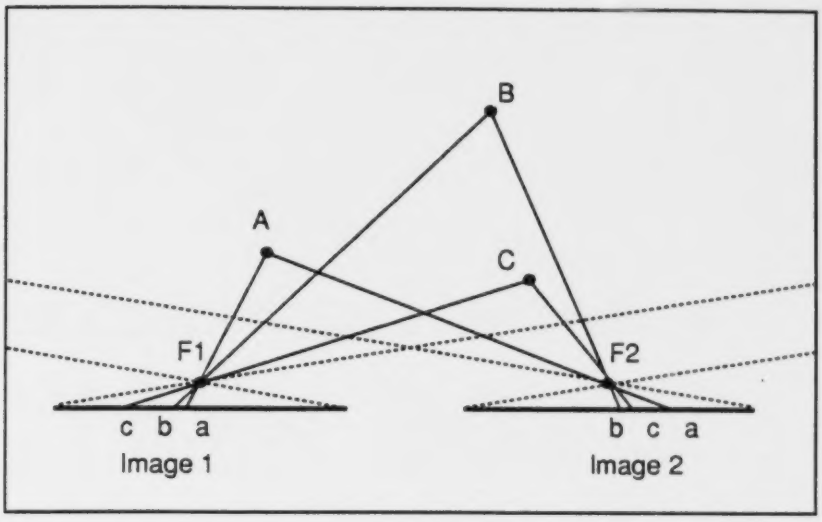

Figure 2.4: Point C violates the ordering constraint.

this situation.

The uniqueness constraint, the continuity constraint, the contrast sign and the ordering constraint along epipolar lines are useful because they do not depend on the specific properties of the scene but on general properties of the stereo geometry. Stereo matching methods in general can be divided into two categories, feature-based and area-based. When interesting features are first selected from the image and the matching algorithm is applied to this subset, a feature-based matching technique is used. Area-based matching is used, when the matched features are low level and dense, and the matching algorithm is applied locally throughout the image at each pixel.

The feature-based approach matches features rather than texture regions in the two images. This approach is tolerant to noise, and can provide precise matching between images. The most commonly used features are points along the edges in the images. This is because edge points are unique and can reduce ambiguous matches between images. However, the feature-based methods provide only coarse matches

and require interpolation. Marr and Poggio [41] first proposed a feature-based stereo matching algorithm using zero crossings in the Laplacian of the Gaussian of the intensity, as the matching features. Later, their algorithm was implemented by Grimson [27]. The stereo matching algorithm was based on the uniqueness and continuity constraints, and matched the corresponding zero-crossings between two images. In their algorithm, a relaxation technique was used to find the matches. This technique was based on 'educated' guesses of the matches found at coarser resolution. They used this information to guide the matching process at finer resolutions. This algorithm has been tested mostly on synthetic images called random dot stereograms, and does not perform well on real-world images. This was because the constraints it uses are not stringent enough to deal with complex scenes.

Pollar, Mayhew and Frisby [42] used edge points instead of zero-crossings. In their algorithm, they used strength(intensity value) and orientation of the edge to enforce the matching. Their algorithm was an iterative winner-take-all procedure. At each iteration, only match pairs with maximum matching strength were chosen. Because the uniqueness constraint was used, all other match pairs related to previously selected match pairs were eliminated from further consideration. This process continues, until all the matched pairs were found. Over 95% of points were matched correctly, with 4.5% matched incorrectly and a 0.5% being left incorrectly unmatched when random dot stereograms were tested [42].

Area-based methods in general are applied to the analysis of aerial images, where the surface varies smoothly and continuously. Area-based matching is based on finding the similarity of corresponding areas between two images. They have the advantage of directly generating a dense disparity map, but they are sensitive to noise and breakdown where there is a lack of texture and where depth discontinuities occur. Forstner [20] used an iterative prediction and correction method in his area-based matching

process. He used edges to verify the prediction. His program was demonstrated only for aerial images.

Hannah [28] used area-based matching in her depth acquisition system. In her approach, the corresponding point was searched on the basis of the similarity of the neighbouring regions defined by a window. The matching point was the center of the window. Points in the window in one image were compared with similarly sized windows of points in the second image. The goodness of each match was determined on difference metrics and correlation measures. In her matching algorithm, the epipolar constraint was used to limit the searching area. For 10 of the 12 tests provided by stereo vision researchers and the photogrammetric community, her system was able to produce correct results. Her algorithm works well most of the time, but difficulties occur when the two images do not contain enough features or when a scene has many depth discontinuities.

2.2.3 Depth Calculation

Once the two corresponding points at (x_1, y_1) and at (x_2, y_2) are obtained in the two image planes, depth determination becomes straight forward. The three-dimensional coordinate information can be calculated based on the Direct Linear Transformation(DLT) or the collinearity equations [56]. The collinearity method is the most straightforward way to model the physical camera system. It describes each component of the physical camera system mathematically, and it is based on an assumption that an object point, a perspective center of the camera, and an image point lie on a straight line. As a result, the collinearity equation can be written as:

$$\overline{\alpha} = \kappa \mathbf{MA} \tag{2.1}$$

Where $\overline{\alpha}$ is a three dimensional image vector. It can be expressed in the image

coordinate system as:

$$\overline{\alpha} = \begin{bmatrix} x - x_0 \\ y - y_0 \\ -c \end{bmatrix} \tag{2.2}$$

with x_0 and y_0 are the coordinates of the principal point, c is the principal distance of the lens, and x and y are the coordinates of a image point. Parameter **A** is a vector from the perspective center to a point in space, and can be expressed in the object space coordinate system as:

$$\mathbf{A} = \begin{bmatrix} X - X_c \\ Y - Y_c \\ Z - Z_c \end{bmatrix} \tag{2.3}$$

with X,Y, and Z are the coordinates of the point, and X_c,Y_c, and Z_c are the coordinates of the perspective center of the camera. As a result, the collinearity equation can be rewritten as:

$$\begin{bmatrix} x - x_0 \\ y - y_0 \\ -c \end{bmatrix} = \kappa \mathbf{M} \begin{bmatrix} X - X_c \\ Y - Y_c \\ Z - Z_c \end{bmatrix} \tag{2.4}$$

Where κ is an unknown scale factor,and in general, it can be eliminated during the evaluation process. **M** is a 3-by-3 orthogonal rotation matrix. It describes the angular relationship between the image and object space coordinate systems. It can be written as:

$$\mathbf{M} = \mathbf{M}_\kappa \mathbf{M}_\phi \mathbf{M}_\omega = \begin{bmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & m_{33} \end{bmatrix} \tag{2.5}$$

The $\mathbf{M}_\omega$, $\mathbf{M}_\phi$, and $\mathbf{M}_\kappa$ matrices are 3-by-3 orthogonal transformation matrices corresponding to successive rotations omega, phi and kappa about the x,y and z axes respectively.

Equation (2.4) yields three linear equations. Dividing the first and second equations by the third equation yields the collinearity equations:

$$f_1 = x - x_0 + c\frac{m_{11}(X - X_c) + m_{12}(Y - Y_c) + m_{13}(Z - Z_c)}{m_{31}(X - X_c) + m_{32}(Y - Y_c) + m_{33}(Z - Z_c)} = 0 \qquad (2.6)$$

$$f_2 = y - y_0 + c\frac{m_{21}(X - X_c) + m_{22}(Y - Y_c) + m_{23}(Z - Z_c)}{m_{31}(X - X_c) + m_{32}(Y - Y_c) + m_{33}(Z - Z_c)} = 0$$

Since equations (2.6) are non-linear, they need to be linearized before being used in a least square solution. The collinearity equations contain the coordinates of the object points, the coordinates and orientation of the camera, and parameters for the interior orientation of the camera. They can be written for each image of each point, and the equations can be combined into a simultaneous solution for the unknown parameters.

Another approach used to model the physical camera system is the Direct Linear Transformation, or DLT, as originally proposed by Abdel-Aziz and Karara [1]. Its derivation begins with the collinearity equation as given in equation (2.6). It does not require initial approximations for the parameters when solving the coordinates of the object points.

2.3 Structured Light Method

Another approach, which is the most common for machine vision applications, is the structured light method. The basic idea is the same as for the conventional stereo vision technique, except that it uses a projector to add patterns to the scene, and

thus increase the number of matching features.This approach simplifies the matching problem, and it can be used on a featureless surface and on irregularly shaped objects.

There are various light patterns which can be used in this technique. Points, parallel lines or a rectangular grid are the most common ones. It does not matter what kind of light patterns are used, image pairs of the object must be recorded. One way to record the image pair is to use a conventional stereo camera set. In this case, the process of matching the corresponding features between two images and depth calculation are similar to stereo vision methods. Another way is to replace one of the two stereo cameras by a projector and thus only one object image is recorded. The half-image of the stereo pair contains complete information about the surface shape if an illuminated pattern with known geometry is used. In this case, the matching process is performed on the object image and the projected pattern on the projector can be points, or parallel lines. The depth information of the 3D object can be calculated by triangulation [46].

2.3.1 Spot Projection

Lewis and Sopwith [38] designed a prototype depth acquisition system using two cameras and a projected dot pattern. In their design, fiducial marks in the form of light emitting diodes were used. The fiducial marks allowed scaling of the data and provide alignment marks during digitization of the dot patterns. There were three main steps in their design. The first step was to extract the coordinates of each spot. This step was achieved by taking the 'center of gravity' of each dot in the image, and storing them in a simple array. The next step was to convert the spot coordinates to a base plane projection coordinate system from the image coordinate system. This involved computing transformation matrices between the two coordinate systems. The final step in their design was the application of the disambiguation algorithm. The algorithm was developed based on stereoscopic constraints similar to those of

the stereo vision matching technique. This system had the advantage of simple image analysis, but, it required a complicated matching algorithm. The matching became more difficult if there was occlusion in the image pair. In the Lewis and Sopwith prototype design, the surface of a cylindrical object was successfully reconstructed, but, application to the medical field, for which the system was originally intended, was not implemented.

2.3.2 Light-Strip Method

Frobin and Hierholzer [23] developed a body surface measurement system using a single camera and a projector with a parallel line pattern. In this system, complete three-dimensional data of the human body surface was reconstructed by using a single image. The process of matching was done on a single object image and a reference image (projected parallel line pattern). Because they used parallel lines as a projected pattern, their system did not require matching point by point between images. Instead it only needed to match the corresponding lines between the object image and the reference image. This reduced the complexity of their matching algorithm. In their design, image processing was broken into several steps. The steps were peak detection, line search, line sequence analysis, and line numbering. Peak detection determined the location of the peak intensity pixel in the line pattern, so that correspondence between images could be performed on one-pixel-thick lines. The line search procedure connected discrete pixels to continuous lines. Line sequence analysis and line numbering arranged and labelled the lines according to their natural topological order, so that corresponding lines between images were matched. The accuracy of the reconstructed surface points on the human body was 0.5 mm perpendicular to the surface(z coordinate). However, this system required the application of complicated image analysis techniques, and the image analysis was compounded by sudden changes in depth in the scene.

2.3.3 Grid Pattern Approach

Dunn and Keizer [15] developed an imaging system to record and measure three-dimensional surfaces of the body. Their system was used to measure the surface area of burns, to measure the volume of facial swelling, and to measure possible expansible tumours. In their system, a projector was used to project a marked grid pattern onto the body surface. An image of this pattern was recorded using only a single camera. There were four required steps in this system: 1) determining the camera calibration matrix and the projector calibration matrix, 2) using image processing techniques to locate grid intersections in the camera image, 3) determining the correct correspondence between labelled intersections on the side and detected intersections in the camera image, 4) based on triangulation method, determining the three-dimensional coordinates of a point on the body surface. Determining the correct correspondence between images and solving the grid labelling problem were the key steps in this system. This system could recover accurate three-dimensional data. The rms error in determining the radius was within 2.8% of the measured radius, when the measured radius is 67.5 millimetres. However, it required a complicated grid labelling schemes. Its camera calibration and projector calibration required human intervention; thus, it might have problems if the system setup is changed. In addition, it required interpolation to produce a surface shape, because a marked grid pattern was used in their system.

2.3.4 Single Camera System Depth Calculation Techniques

David Elad et al. [16] developed a scheme to calculate three-dimensional information in their single camera system. Their three-dimensional data evaluation scheme did not follow the usual procedure of standard photogrammetry. Their reconstruction method was based on innovative image processing procedures. In their system, calculation the actual values (X,Y,Z) of the measured surface was based on assumptions that 1)

the planar (x,y) dimensions of the image were linearly related to the actual planar dimensions (X,Y) of the object, and (2) the heights (Z) of points on the measured surface were linearly related to their projection on the plane of the image. Thus, calculation of X and Y was done with the aid of known control points. The height (Z) of each point was evaluated by comparing the distortion of each line of the object with the line on a control plane. This reconstruction method was simple, and provided a good method of solving the reconstruction problem. However, to get accurate 3D information, this method required each illuminated line pattern to perpendicularly cut the control planes. In other words, it required a relatively large distance between the projector and the object, as compared with the heights of the measured object.

2.4 Camera Calibration Techniques

One of the main task of a 3D measurement system is to get the 3D coordinates of points from information on the corresponding image coordinates. As a result, understanding the transformations involved in mapping 3D points to 2D image points becomes important. In general, camera calibration problems can be seen as the determination of interior and exterior orientation camera parameters. There are many different camera calibration schemes. Different camera calibration scheme involve different combinations of known and unknown parameters, but the fundamental mathematics of them remain the same [34]. Terminology important to the camera calibration process is given in the following sub-section.

Intrinsic Parameters:

Interior orientation refers to two sets of parameters. The first set is the geometric parameters of the camera itself, such as the principal distance and the principal point coordinates. The other set is the parameters that describe the various distortions. The most common distortions are de-centering error, and radial distortion of the lens. The

problem of line synchronization between scanning and acquisition is another concern in calibration of CCD-cameras [24].

Extrinsic Parameters:

The exterior orientation of a camera refers to camera position and orientation relative to the three-dimensional spatial coordinate system. There are a total of six extrinsic parameters for each camera. They are X, Y, Z coordinates and ω, ϕ, κ three rotation angles.

If the exterior orientations of the cameras are known, and the point coordinates are unknown, the image rays from each camera position can be intersected to obtain the 3D coordinates. This is known as the 'intersection problem'. If the exterior orientations of the cameras are unknown or are partially known but enough point coordinates in the object space are known, the orientation parameters can be determined. This is referred to as the 'resection problem' [33].

2.4.1 Standard Photogrammetric Methods

Determination of the interior orientation of the camera is the first step in camera calibration. It involves determination of principal distance, principal point coordinates and lens distortion. In the ideal situation, an object point, its corresponding image point, and the camera's principal point lie on a straight line(collinearity condition). Thus, principal point coordinates can be obtained from the imaging geometry, if enough corresponding points between images and the real world are known. However, due to manufacturing tolerance, the principal distance is normally different from the focal length marked on the lens [8]. In addition, the image point coordinates usually have an offset correction factor(δ_x, δ_y). According to Peterson et al. [47], the

correction factor can be modelled as follows:

$$\delta_x = (x - x_p)(K_1\delta + K_2\delta^2) + [P_1(\delta + 2(x - x_p)^2) + 2P_2(x - x_p)(y - y_p)] \quad (2.7)$$

$$\delta_y = (y - y_p)(K_1\delta + K_2\delta^2) + [P_2(\delta + 2(y - y_p)^2) + 2P_1(x - x_p)(y - y_p)]$$

with

$$\delta = (x - x_p)^2 + (y - y_p)^2.$$

In these equations, K_1 and K_2 are radial distortion parameters, P_1 and P_2 are parameters for decentering distortion, and (x_p,y_p) are the principal point coordinates. These set of equations are non-linear and can be linearized using a Taylor Series expansion.

The determination of the exterior orientation of cameras can be done using the the collinearity equations described in section 2.2.3. By knowing the 3D coordinates of a minimum of six object points, and the coordinates of the corresponding points in the two images, a total of 12 equations is produced. Using these 12 equations, the 12 unknown of cameras' exterior orientation parameters can be calculated. The collinearity condition method can be used to calibrate the camera system.

The coplanarity equation [56] can be used, when the interest is in the relative orientation of two cameras. The coplanarity equation is an extension of the collinearity equation. For the collinearity condition, the object point, perspective center of the camera, and the image point are required to lie on a straight line. Whereas, the coplanarity equation assumes that two the images points from the object point, the two camera perspective centers, and the object point itself all lie on a single plane. Thus, the coplanarity condition can be expressed mathematically. The coordinates of the object point do not appear in this mathematical expression, so no approximations for the coordinates are needed. The equation is based on the vector between the two perspective centers, and the image point vector on each image. It contains

12 unknowns, which include three coordinates and three orientation angles for each of the two cameras, and it assumes that the interior orientation of the cameras is known.

2.4.2 Single Camera and Single Projector Calibration Technique

The lack of having a set of image coordinates from two cameras for control points due to the fact that only one camera is used to record information causes a major problem in the camera calibration. In addition, because the projector does not record object information, calibration of the projector can not be performed directly as with standard camera calibrations. However, by implementing certain procedures, calibration of a camera and a projector can still be done.

Frobin [22] , calibrated a video system based on standard photogrammetric methods. However, to apply the standard calibration methods, special procedures of image data acquisition were required. Standard photogrammetric techniques use control points recorded in a stereo image pair to solve the perspective matrices. If the spatial coordinates of the control point pair were determined a priori, the interior and exterior parameters of the cameras could be determined. The problem was that only one half-image(one of stereo pair images) of the control point system was recorded. The other half-image was produced by the projector and did not exist as a fixed record. However, a substitute projector half-image could be calculated from data measured in the camera image. Frobin calculated the coordinates of the control points of the projector half-image by interpolating the projected lines on the camera image. After formating the stereo image pair, the collinearity equations was used in their design.

Dunn and Keizer [15] developed an automatic camera and projector calibration scheme in their system. In their scheme, the camera was calibrated first. To calibrate

the camera, a picture was taken of a marked grid glued to the side of a box and placed at a known position in the camera's field of view. After taking an image of the marked grid and applying image processing techniques, the two-dimensional locations of the intersections in the image plane were determined. The three-dimensional locations were determined from the grid labels, with a known scale of the pattern and known position of the box in the three-dimensional coordinate system. The camera parameters were then computed.

The projector calibration matrix was determined next. To determine the projector matrix, the picture of the marked grid on the box was replaced by a blank piece of paper, and the box was placed at a known position. A slide with grid pattern was placed in the projector and projected onto the box. The first camera image was saved. The box was then moved to a second position, and a second image was taken. After processing each image, the two-dimensional locations of the intersections and their grid labels were determined. From the two-dimensional locations and the camera calibration matrix, each intersection was back-projected to its location on the box in the three-dimensional coordinate system. This was repeated for each intersection on both projector images. From the grid labels and three-dimensional locations, the projector calibration matrix was computed.

2.5 Section Summary

Three different optical techniques (scanning method, stereo vision method, and structured light technique) for finding three-dimensional information were discussed. The depth calculation techniques and the camera calibration procedures associated with the three different techniques were described. The scanning method can produce accurate three-dimensional coordinates. It does not require any image processing and matching algorithms and it can be used to build a highly automated three-dimensional

acquisition system. However, the major drawback of the scanning system for biomedical applications is the excessive time required to capture the three-dimensional data of a full trunk. Any patient posture changes due to sway and breathing introduce unacceptable levels of artifacts. Stereo photogrammetry can also produce accurate and reliable three-dimensional information; however, it has not been considered for most medical applications [44]. This is because the technique depends heavily on ambient light; a change in lighting conditions causing change of pixel intensity, will introduce inaccuracies. Moreover, it requires complex feature extraction and matching algorithms, especially, when the scene is of a smooth and featureless body surface. The structured light method is attractive, because it is particularly suited to measuring a object surface shape when physical contact is undesirable, when the measured object surface is featureless, and when the image acquisition process takes place in a room environment. In addition, this technique can produce reliable three-dimensional results and can record images in a short time. Based on these advantages, the structured light technique was selected for the video system developed in this research.

Chapter 3

SYSTEM DESIGN AND CALIBRATION

3.1 Apparatus

Many structured light systems have been developed for biomedical applications. Dunn et al. [15] applied structured light for the measurement of surfaces for burn victims, Frobin and Hierholzer et al. [21] developed a structured light system for monitoring the progress of scoliosis and Gregory et al. [26] used a structured light system for the planning and evaluation of facial surgery. These systems consist of a one projector and one camera. They require calibration of the projector. The proposed system uses two cameras and is based on stereography principles. It requires only standard camera calibration and reconstruction procedures. As a result, it simplifies the three-dimensional data acquisition process.

3.1.1 Video System Design

The basic design of a three-dimensional acquisition system is similar to the systems used in stereo photogrammetry. In figure 3.1. a schematic outline of the system is given. It uses two cameras and a projector with parallel line patterns on a standard 35mm slide. Two JVC TK-S210U CCD video cameras (JVC, Japan) with 16mm f8 lenses are used. The cameras are mounted symmetrically above and below the projec-

tor. The base line distance between the two cameras is approximately 1086mm, and the horizontal distance between the cameras and a subject is approximately 1613mm. Convergence angles of approximately 19° are set for both video cameras. It is found that with these convergence angles and distances, both cameras will approximately have the same field of view in the object space, and are able to capture a full human trunk surface. A Scion Image Capture II board(Scion Corp., Walkersville, USA) installed in a Macintosh II computer (Apple Computer, Inc., Cupertino, USA) is used to capture subject images. The sensor array of the cameras is approximately 8.8mm by 6.6mm and contains a grid of 649 by 491 pixels. The capture board yields 640 by 480 samples with 8 bit gray level resolution images. The spatial resolution of the cameras is approximately 1 pixel per millimetre. The horizontal:vertical aspect ratio of the cameras is found to be about 1:1.03. To make the best use of the cameras and the frame grabber resolution, the cameras are rotated by 90° (portrait format). Thus, the scan lines run vertically, corresponding to the X coordinate in the object space and are the rows of the pixel image.(see Figure 3.1).

The projector lens parameters are not used in the calculation of three-dimensional coordinate information; a f2.5/3.55mm Hi-Speed Super-wide projector lens (BUHL Optical co., Pittsburgh) was chosen so that the lens would produce a larger projected pattern than the cameras' field of view. To simplify the system setup, the optical axis of the projector is positioned horizontally, and the projector slide is approximately parallel to the standing subject. Because image data acquisition is completely independent of the analysis, a real time operating system is not required. The system software was developed on Unix IBM RISC 6000, and implemented in the C programming language.

During image data acquisition, the measured subject is positioned against an object stand. A black curtain is placed behind the subject to increase the contrast of the image. After the projector has been turned to project a line pattern onto the

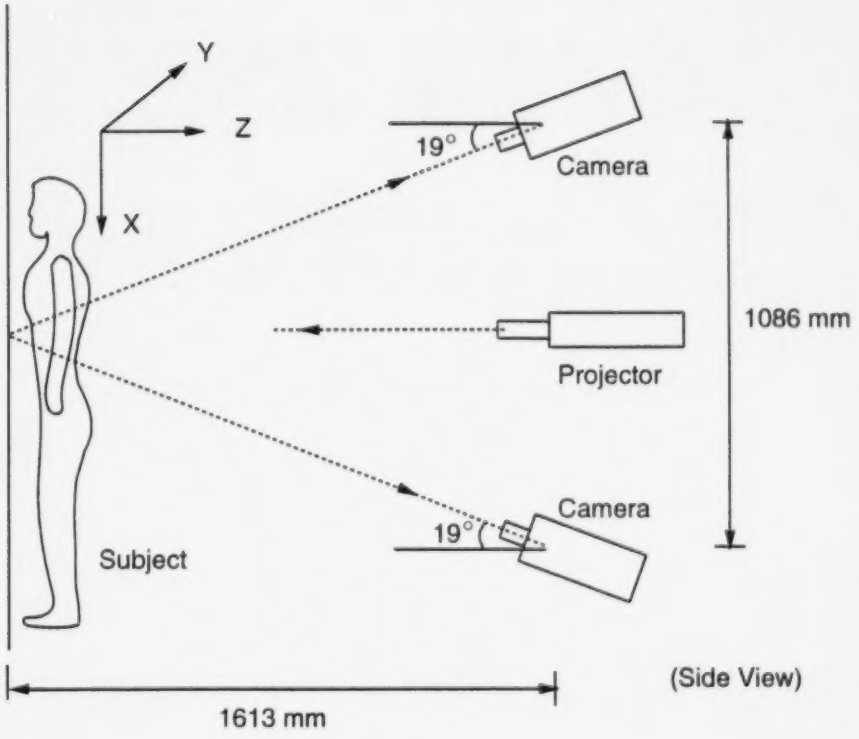

Figure 3.1: Structure light system with three optical components.

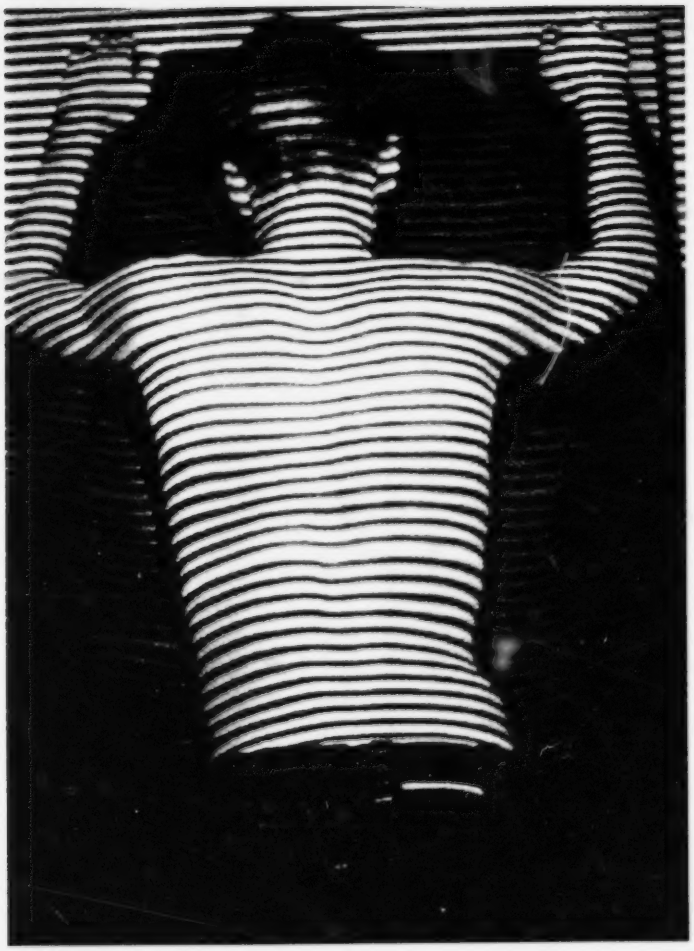

Figure 3.2: Parallel line pattern projected on a body surface.

subject, the room is darkened, the aperture of both cameras are adjusted, and digital images from both cameras are recorded under computer control. The purpose of the object stand is to ensure that the measured subject surface is located within the field of views of both cameras and the projector. The field of view of the cameras is approximately 595mm by 785mm. Figure 3.2 shows a typical camera image captured by the video camera system.

3.1.2 Projector Slide Design

Lines projected onto the subject surface are encoded in a pattern of thin and thick lines. Figure 3.3 shows a picture of the coded line pattern. Using the center thick line as an origin, the thick lines are located at 0, ± 5, ± 15, ± 25 line units. The line width for the thin and thick lines on the image template of the projector slide is 1 pixel and 2 pixels, respectively. The space between the lines is 4 pixels. This particular line pattern yields acceptable data resolution for the subject surface, and at the same time does not produce false line interconnections when it is projected onto the subject surface. The density of this line pattern when imaging the subject surface is approximately 1 line per 1.3 centimetres.

There are two advantages from the projection of a coded parallel line pattern as opposed to other patterns. Firstly, object surfaces, especially in the medical field, are often featureless. Projecting a coded line pattern on the object surface adds features to the surface images, and allows three-dimensional measurements to be made. Secondly, the connectivity of the line pattern and the special relationship of the thin and the thick lines in the coded line pattern simplifies the pattern recognition procedure, and reduces the complexity of the stereoscopic matching process.

To develop the desired image template of the line pattern, a program (written in the C programming language) was designed and implemented. This program was used

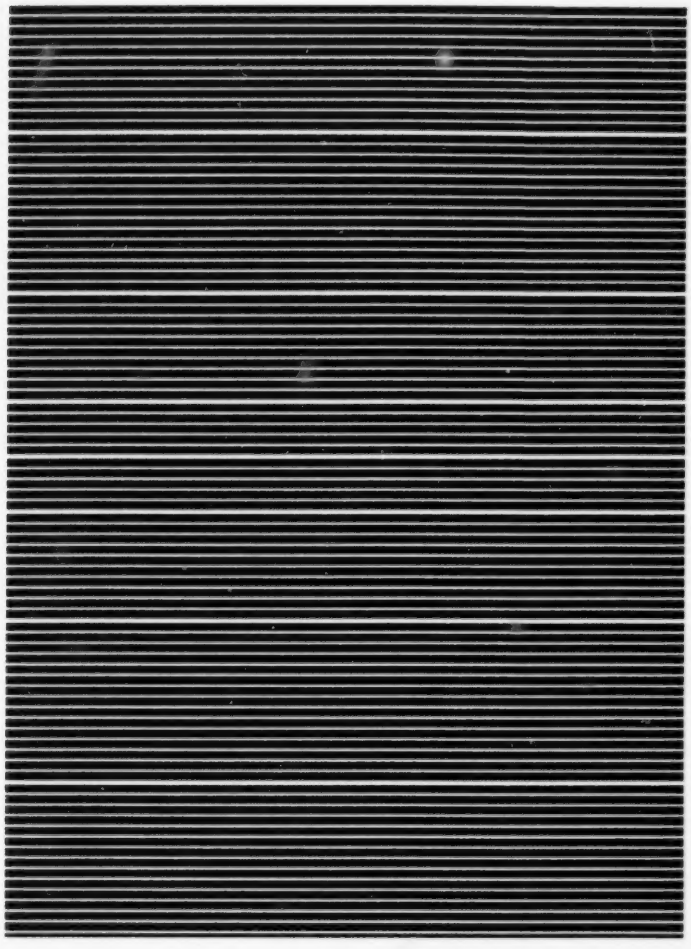

Figure 3.3: A image of coded line pattern.

to develop an image template printed directly on the projector slide. The program contains two subroutines. The first subroutine defines the location of the lines and draws the lines according to a specified width. The second subroutine converts an image array into tiff image array format. and outputs a tiff image file.

3.2 Video Camera Calibration

Camera calibration is used to determine the interior and exterior orientation parameters. The bundle adjustment method of photogrammetry is used [47]. This method is based on the collinearity condition and assumes that the perspective center for the camera, the image point, and the point in the object space all lie on the same straight line. The details of the collinearity equation were presented in section 2.2.3.

3.2.1 Camera Calibration Control Field

Figure 3.4 shows an outline of an object stand with a control frame. The object stand is 934mm high, and is mounted at a distance of 1613mm from the projector. The control frame, 635mm by 762mm, is placed vertically on the object stand during video camera calibration. The control frame consists of thirty control points with known X, Y, and Z coordinates, and numbered from 1 to 30 starting from the top right to the bottom left. This sequence is used during the calibration of the video cameras. It ensures the correspondence between the known three-dimensional coordinates and the image coordinates of the control points. Position number 25, which is the top left corner of the control frame, is defined as the origin of the control system, and the X-Y-Z coordinate reference frame for the system is defined as depicted in figure 3.4.

To calibrate both cameras in the system, the control frame is positioned vertically on the object stand. The X, Y and Z coordinates of the control frame were measured by a numerical milling machine, and are as given in table 3.1. The precision of the digital read out of the milling machine is 0.0254 millimetres(0.001 inch). The control frame images from both cameras were recorded under computer control. Figure 3.5 shows one of the control frame images. As shown in figure, the thirty control points visible in the image are at different depths to enhance the solution of the system of equations, (Eq. 2.6), that determine the camera's location and orientation, and ensures that the solution of calibration parameters are reliable.

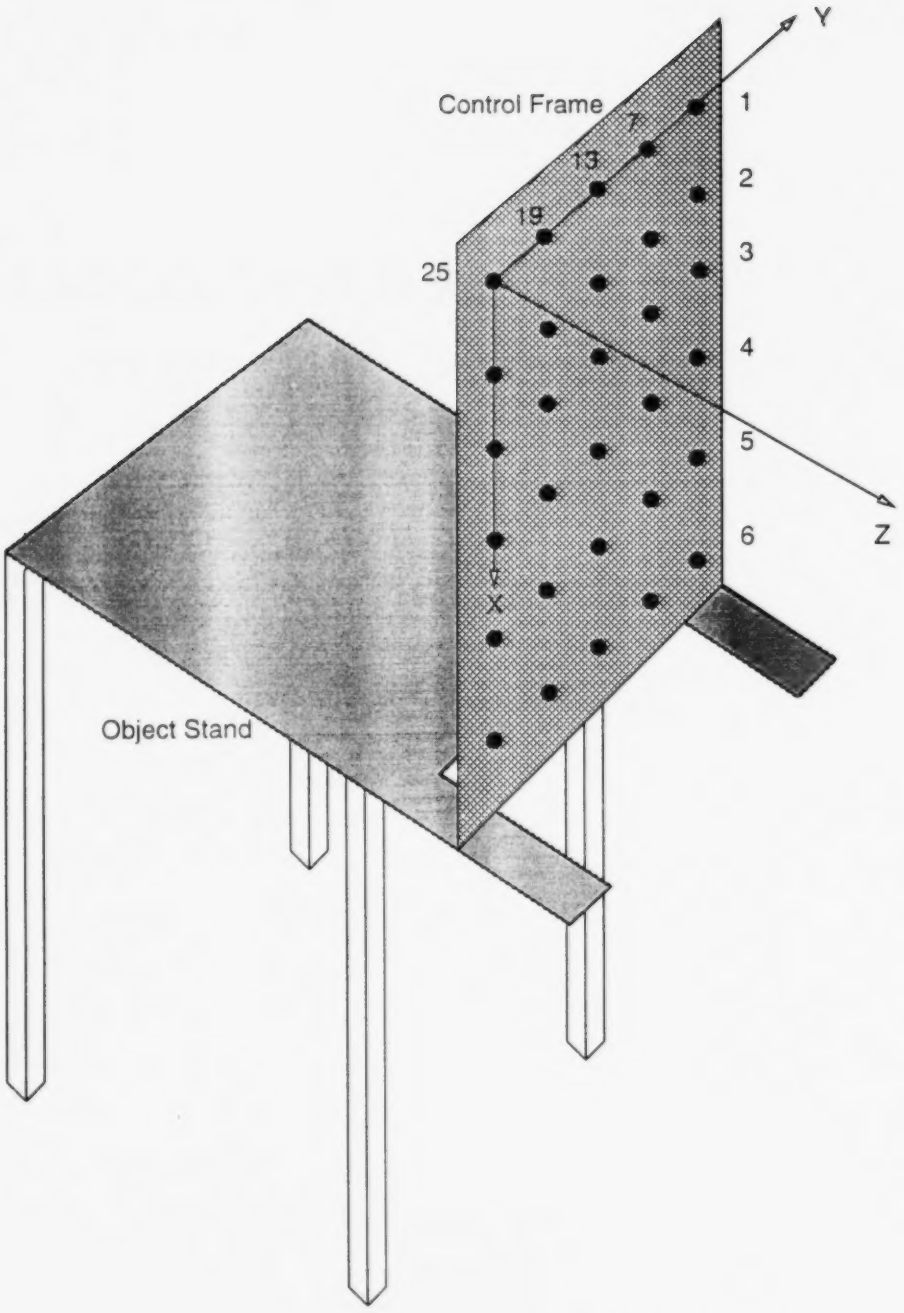

Figure 3.4: A control frame mounted on a object stand.

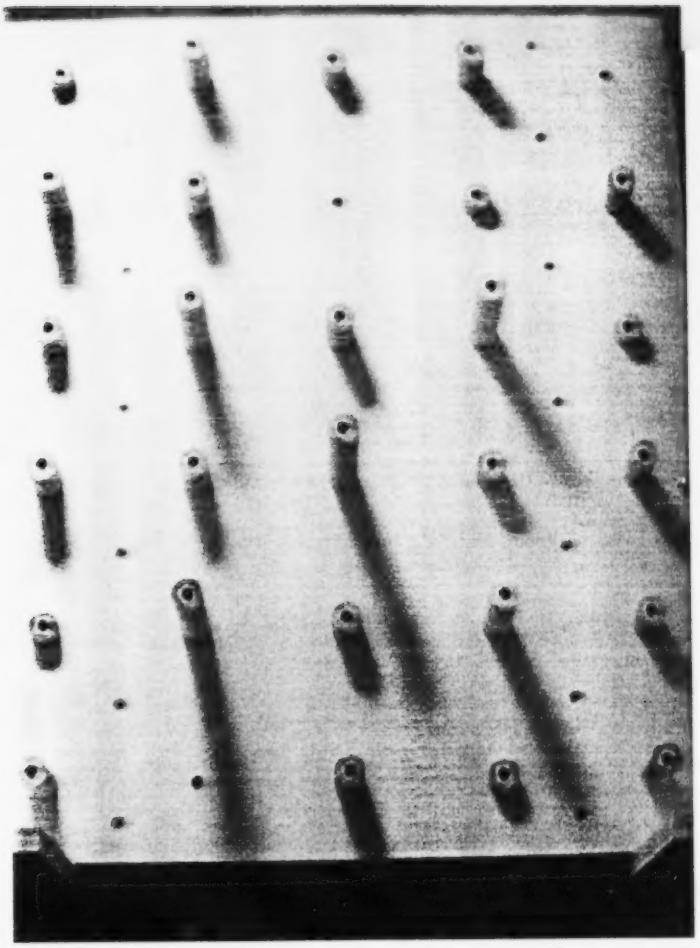

Figure 3.5: Image of the control frame.

During camera calibration, the x-y image coordinates of the control points were manually determined in each image and manually correlated with each image. Since the three-dimensional coordinates of each control point was known, and once both sets of x-y image coordinates were established from the images, the exterior and interior parameters of the cameras could be computed.

Point	X (mm)	Y (mm)	Z (mm)
1	0.0	508.0	0.0
2	127.0	508.0	44.9
3	254.0	508.0	14.9
4	381.0	508.0	44.8
5	508.0	508.0	29.8
6	635.0	508.0	59.9
7	0.0	381.0	44.8
8	127.0	381.0	14.9
9	254.0	381.0	104.9
10	381.0	381.0	29.9
11	508.0	381.0	89.8
12	635.0	381.0	14.7
13	0.0	254.0	29.9
14	127.0	254.0	0.0
15	254.0	254.0	45.1
16	381.0	254.0	120.2
17	508.0	254.0	30.1
18	635.0	254.0	44.9
19	0.0	127.0	60.0
20	127.0	127.0	45.1
21	254.0	127.0	90.2
22	381.0	127.0	45.0
23	508.0	127.0	105.0
24	635.0	127.0	0.0
25	0.00	0.0	15.1
26	127.0	0.0	60.2
27	254.0	0.0	30.1
28	381.0	0.0	44.9
29	508.0	0.0	15.0
30	635.0	0.0	59.9

Table 3.1: Control point XYZ coordinates.

3.2.2 Calibration Procedure

To calibrate the cameras accurately, an available camera calibration program(**V2STEREO**) written in FORTRAN and based on Eqs. 2.6 was used (A. Peterson Dept. of Civil Eng., U of A). The collinearity equations and the least squares solution method are the basis for the program. The program has an option that allows for parameters such as radial distortion, K_1, K_2, decentering distortion, P_1, P_2 and the principal point coordinates, (x_p,y_p) to be computed during camera calibration. The program requires the input of the X-Y-Z coordinates for each control point, both sets of x-y image coordinates of the control points, and an approximation for the cameras' exterior and interior parameters. The approximated exterior and interior parameters of both cameras are obtained using a preliminary program(**DLT**). This program computes the camera's exterior and interior parameters using an image to object coordinate transformation based on the Direct Linear Transform (DLT) equations [56]. The preliminary program also requires the input of X-Y-Z coordinate information and a set of x-y image coordinates for each of the control points. The x-y image coordinates for the control points are generated with a program called **RCCD**. The input of this program are the pixel coordinates (rows and columns) of the control points. In the program, the center of the camera grids is set at x-pixel = 284 and y-pixel = 208 with an origin being in the lower left hand corner. Based on this image center, the collected pixel data is converted into image centered x-y image coordinates. In addition, based on the specified resolution of the camera, the centered x-y image coordinates of each control point are scaled to millimetres for convenience. Detail of camera calibration procedures are presented in Figure 3.6.

3.3 Calibration Experimentation

The calibration program(**V2STEREO**) solves for thirteen parameters of both cameras in the system. They are X, Y, and Z coordinates of the camera's perspective

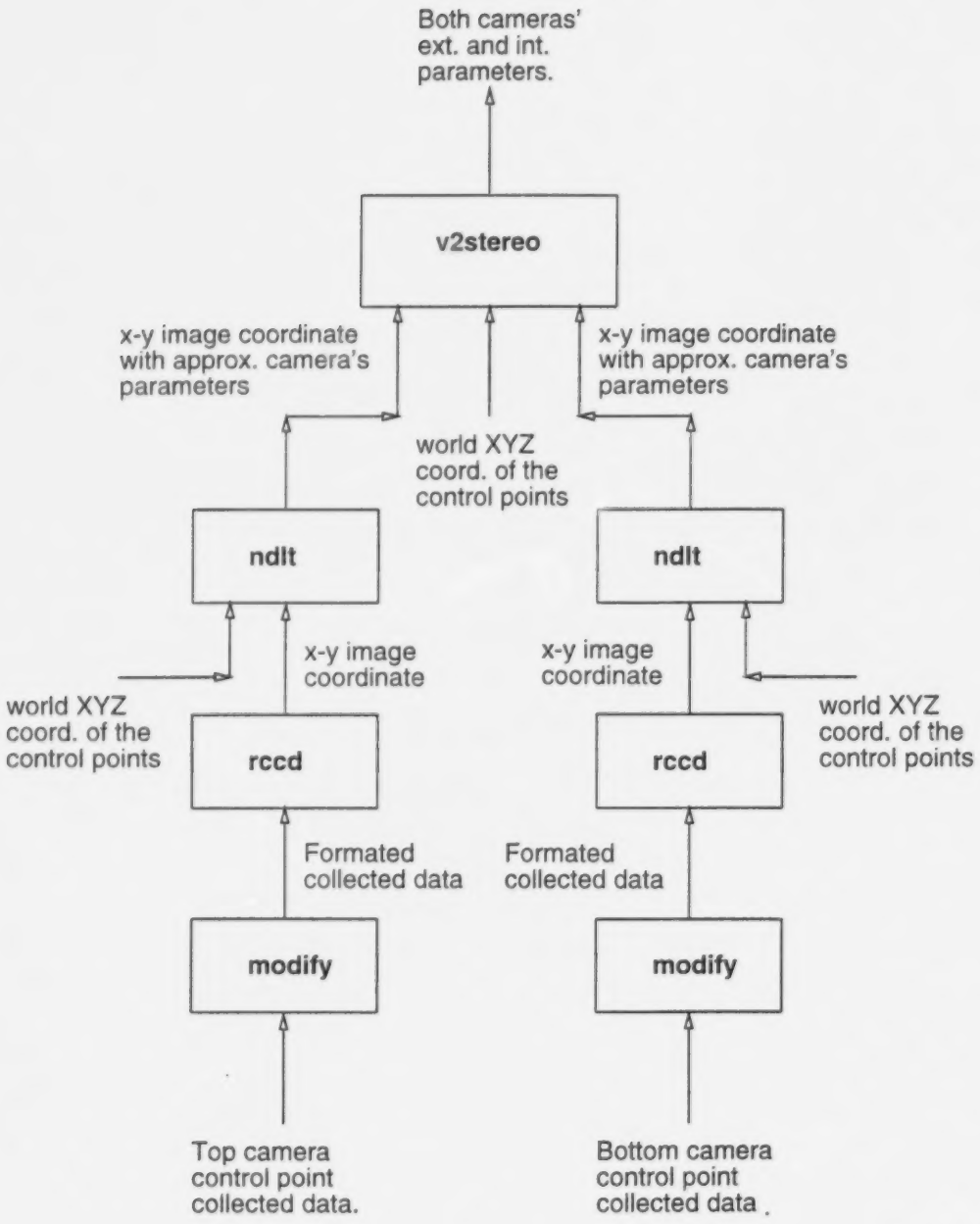

Figure 3.6: Camera calibration procedures.

Result	Camera	X_o mm	Y_o mm	Z_o mm	$Focus$ mm	ω deg	ϕ deg	κ deg
1	Top	-201.94	240.14	1586.43	17.22	0.34	-19.19	-0.31
	Bottom	909.09	242.94	1590.27	16.56	0.66	18.48	-0.59
2	Top	-202.33	241.35	1635.78	17.30	0.31	-19.57	-0.30
	Bottom	906.46	241.88	1574.77	16.38	0.67	18.57	-0.98
3	Top	-199.19	239.02	1621.32	17.15	0.39	-19.16	-0.33
	Bottom	916.47	242.51	1603.50	16.73	0.64	18.58	-0.64
4	Top	-209.46	241.18	1636.11	17.35	0.31	-19.32	-0.34
	Bottom	916.53	242.24	1596.15	16.64	0.66	18.67	-0.59

Table 3.2: Four sets of camera calibration results.

center, the orientation of the camera (rotation angles ω, ϕ, and κ with respect to the X, Y and Z axes), the focal length, two radial distortion parameters, K_1, K_2, and two decentering distortion parameters, P_1, P_2 as well as the principle point coordinates(x_p, y_p). However, during camera calibration, the solution to the system of equations, Eqs 2.6 diverges if system is susceptible to parameter linkage(high correlation) leading to ill-conditioning of the equation system. By fixing the principle point location, the solutions of the system of equations converge easily(four passes).

3.3.1 Calibration Results

To check the repeatability of the camera calibration, four sets of control frame images were taken. Replications of the control frame images were taken after removing the control frame from the object stand and then repositioning it. This yields eight separate sets of x-y image coordinate data. Four sets of the camera calibration results are presented table 3.2.

As the tables show, the camera calibration results are consistent aside from the camera location(X_o,Y_o,Z_o). The maximum variations in the three orientation angles of the top camera are 0.08°, 0.41° and 0.04°, respectively. The maximum variations in the three orientation angles of the bottom camera are 0.03°, 0.19° and 0.39°,

respectively. The maximum variation in Z_o for the top camera is 50 millimetres, and the maximum variation in Z_o for the bottom camera is 29 millimetres. These differences in the X_o, Y_o, Z_o, and the three camera rotation angles were caused by the differences caused by movement during repositioning of the control frame. To test the validity of the calibration values, the three-dimensional coordinates of the control points were reconstructured using the obtained calibration values. The coordinates obtained in each test were compared to the known coordinates (Table 1) after a seven parameter similarity transformation(3D similarity transformation) [43] to adjust for the different scale and orientation of the obtained coordinates, the results from the four tests gave good agreement with the measured control coordinates. The largest discrepancy of the four test results was 1.9 mm in a Z coordinate with standard deviation at 0.91. This indicates that the results of the camera calibration are reliable and consistent. The calibration results presented here are used in image rectification and three-dimensional data reconstruction.

4.1.1 Region Growing

Region growing [32] is a procedure that groups pixels or subregions into larger regions. Pixel aggregation [7] , one of the simplest approaches is used. This approach starts with a 'seed' point, and grows to include neighbouring pixels having similar properties. There are two steps in this method. They are: 1) select the initial seed that properly represents the region of interest, and 2) select a suitable inclusion criteria in various regions during the growing process. In this application, the central point of the brightest region on the trunk is chosen as the seed point. This point lies on the object and its intensity value is used to represent the entire region of object.

After the seed point in the object is found, the image shown in Figure 4.1 is partitioned into four parts. The region growing process starts at the seed point and moves toward the boundaries of the object in each of the partitioned regions. It includes all candidate points with similar properties. There are two steps in the region growing (pixel aggregation) process. First, the mean intensity value of a 3 by 3 mask with the seed point at the center is calculated, and similarly, the mean intensity value of a 3 by 3 mask with a candidate point at the center is found. To determine the inclusion of the candidate point, the absolute difference in mean intensity value between the seed mask and a candidate mask is computed. If the result does not exceed 50 percent of the difference between the minimum and maximum intensity values in the entire image, and the mean intensity value of the candidate mask is not equal to zero(black), the candidate point is included in the region of interest and becomes the new seed point. A threshold value of 50 percent of the difference between the minimum and maximum intensity values is chosen. It is based on the fact that when a black curtain is placed behind the measuring trunk surface, the intensity value of the black curtain is low, and as a result, the contrast of the trunk surface is greatly enhanced. To determine the inclusion of additional points, the 3 by 3 mask is moved one pixel at a time to the new candidate point location and the mean intensity

Chapter 4

IMAGE FEATURE EXTRACTION

Extracting features from the images is the first step in the three-dimensional acquisition process. Feature extraction defines sets of lines in the two camera images. The sets of lines are used to determine correspondence between the two images. Correctly extracting and identifying the one-pixel-thick lines provides a one-dimensional match between the two images, and is an essential stage in the three-dimensional acquisition process.

4.1 Image Segmentation

Defining the region of interest is the first step in the feature extraction process. To extract the region of interest from the projected line image the boundary of the trunk must be detected. A pair of images are recorded with the subject illuminated by a spot light (slide projector without a slide) in a darkened room. This image (figure 4.1) is used to define the boundary of the region of interest limiting the computation of the three-dimensional coordinates information to the trunk only. Region growing and boundary recognition methods are used to find the boundary of the trunk.

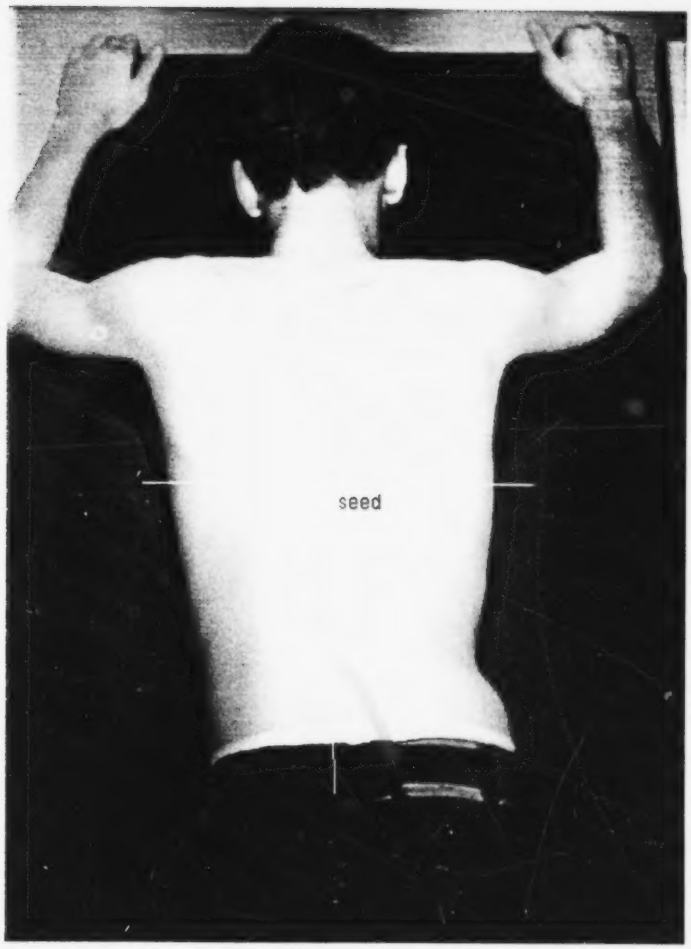

Figure 4.1: Object image with a spot lighting.

value of the mask is calculated. The comparison in the mean intensity value is done between the new candidate mask and the newly updated seed mask. This process is repeated in the four partitioned regions, and terminates when the region of interest is completely defined in the image. Based on the above procedures, the region of interest is defined.

Figure 4.2 shows the result of a typical region growing process. It contains only the object data. The information from the background has been removed. However,

Figure 4.2: Resultant image after applying the region growing technique.

Figure 4.3: An example when region growing process fail.

if the contrast of the trunk surfaces is not enhanced, the region growing process may fail. Figure 4.3 shows an example when the region growing process fails. In this figure, the intensity values of both the gown and the trunk surface are similar. The pixel aggregation procedure can not isolate the region of the trunk. Figure 4.4 shows the resultant image after the pixel aggregation procedure. It contains not only the trunk but also the gown covers on the trunk. Detailed discussion of this image is given in a later chapter.

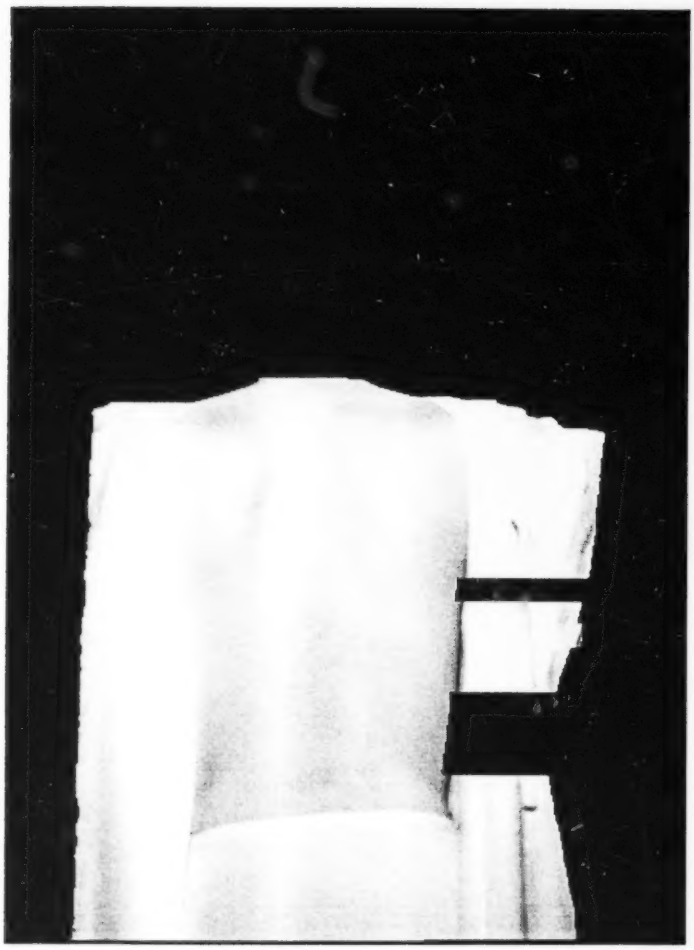

Figure 4.4: Resultant image after pixel aggregation procedure.

4.1.2 Edge Detection

An edge is the boundary between two regions with relatively distinct intensity properties. The computation of a local derivative is used in most edge detection techniques. The local derivative is negative when the transition between two regions is from light to dark and is positive when the transition region is from dark to light. In areas of constant intensity level, the local derivative is zero. The edges in the image can be detected using the magnitude and sign of the first derivative,

The gradient of the pixels in the image is calculated by Sobel operators [50] . Its masks are shown in Figure 4.5. From Figure 4.5, the derivatives based on the Sobel operator masks are:

$$G_x = (z_7 + 2z_8 + z_9) - (z_1 + 2z_2 + z_3) \tag{4.1}$$

and

$$G_y = (z_3 + 2z_6 + z_9) - (z_1 + 2z_4 + z_7) \tag{4.2}$$

The magnitude of the gradient vector of a point is:

$$G = ((G_x)^2 + (G_y)^2)^{\frac{1}{2}} \tag{4.3}$$

and the direction is:

$$\alpha(x, y) = \tan^{-1} \frac{G_y}{G_x} \tag{4.4}$$

where the z_i is the intensity value of pixels in the image.

Using Eq. 4.3, the gradient at the location of the center of the masks is calculated. To get the next value, the mask is moved to the next pixel location and the calculation is repeated. After the computation has been completed for the whole image, the result

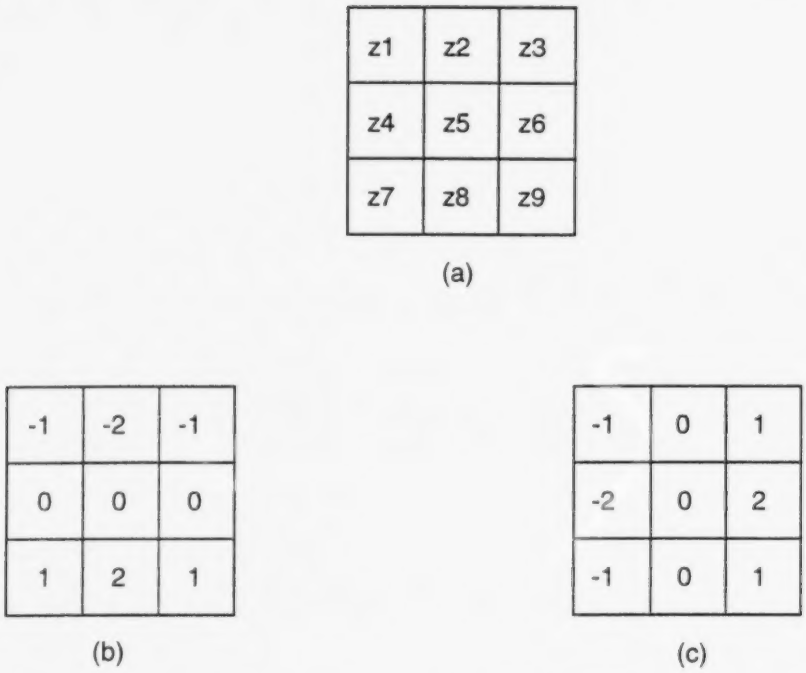

Figure 4.5: Sobel masks: (a) 3 by 3 image region; (b) mask used to compute Gx; (c) mask used to compute Gy.

is a gradient image which is the edges of the measured object. However, because of nonuniform illumination of the image, the boundary of the object is often incomplete; thus, an edge linking procedure is needed to complete the boundary.

In the edge linking procedure, two properties are used for establishing connectivity of edge pixels; 1) the strength (intensity value) of the edge pixels, and 2) the direction of the gradient vector. The first property, defined by Eq 4.3, ensures that the magnitude of the linked edge pixels are similar, and their absolute difference should be less than a threshold. After many trials, a threshold of 40 percent of the difference between the minimum and maximum intensity values in the image was chosen. It was found that this threshold value is optimal for the application because when using a lesser value, the edges of the trunk starts to erode and when using a greater value, the trunk image tends to become noisier. The noise is an unwanted edge produced

when skin pigmentation changes and when there is depth modulation in the trunk surface. The second property, given by Eq. 4.4, ensures that the orientation of the edge pixels are similar, and the absolute difference between angles of two adjencent edge pixels does not exceed an angle threshold. Based on this property, while scanning horizontally for the next connected edge pixel, edge pixels that are only within a region of ± 40 pixels in the horizontal direction from the previous edge pixels are considered. Any edge pixels that are outside of the search limit are discarded. The search region of ± 40 pixels in the horizontal direction is chosen arbitrarily. It is used to limit the search area and ensures that all the connected edge pixels have similar orientation.

Figure 4.6 illustrates the complete boundary of the region of interest. The noise due to variation in skin pigment and depth modulation in the trunk surface has been removed.

4.1.3 Segmentation of The Projected Line Image

After the boundary of the object is defined, the region of interest in the projected line image (figure 3.3) is obtained. This is accomplished by superimposing the trunk boundary onto the projected line image. Figure 4.7 shows the result of this process. As shown in the figure, all data external to the object boundary in the projected line image is discarded, and only information within the object boundary is considered for further processing. This reduces the size of the working matrix.

However, if patients move during the image recording process, the boundary will define the region of interest incorrectly. Consequently, some regions of the trunk are omitted, and the system tries to process lines that are off the body. As a result, unwanted information is generated. Because the pair of the projected line images and the pair of the subject images under a spot light are captured within 1-2 seconds by

Figure 4.6: Enclosed boundary of the trunk.

Figure 4.7: Segmented projected line image.

the pair of the cameras, involuntary patient movement due to breathing and posture sway usually is small. This amount of movement does not normally affect performance of the system or the reliability of the data points the system generates.

4.2 Extraction of One-Pixel-Thick Lines

Extracting one-pixel-thick lines from the segmented line image involves determining the location of the projected lines. An image thresholding technique and line thinning

algorithm are used to determine the one-pixel-thick lines.

4.2.1 Image Thresholding

Thresholding is one of the important techniques that can be used to extract image features [48]. It is useful when there is high contrast between the region of interest and the background. Images with high contrast can be divided into two intensity modes. To extract the objects from the background of the images, a threshold (T) can be selected which separates the two modes. Image thresholding is accomplished by scanning the image pixel by pixel and labelling each pixel as object or background, depending on whether the intensity value of that pixel is greater or less than the threshold value. The difficulty in image thresholding is the selection of a 'good' threshold value. The optimal thresholding value will vary from image to image and from region to region in a image and it depends on the brightness of the image and the reflectance of the object itself. The thresholding value can be viewed as a function and has a form:

$$T = T[x, y, p(x, y), f(x, y)] \tag{4.5}$$

Where f(x, y) is the intensity value of point (x, y) and p(x, y) are the local properties of this point. If T depends only on f(x, y), the threshold is called global. If T depends on both f(x, y) and p(x, y), the threshold is referred to as local, and if T depends on both f(x, y) and spatial coordinates x and y, the threshold is called dynamic.

To enhance the contrast of the projected line in the image, a local dynamic thresholding technique is used. In this technique, the projected line image is partitioned into sub-images and a threshold is determined for each of the sub-images. A 10 by 10 pixel window for local thresholding is chosen because the distance between the

centers of the two projected lines is approximately 10 pixels. The mean intensity value of each sub-image adequately separates the object (the projected lines) from background (the object). This approach is used because the brightness of the projected lines varies as the reflectance and the depth of the object surface varies. It is difficult to find a single threshold T value that can separate the object from the background for the entire projected line image. The local dynamic threshold method, on the other hand, segments the object from the background locally; thus, it not only achieves separation of the object from the background but also compensates for the variations in the reflectance of the surface.

Figure 4.8 shows the projected line image after local thresholding. The contrast of the projected lines is enhanced. The background noise between the projected lines has been removed. The projected lines become well defined in the image.

After thresholding, the projected line images are converted to binary images. The background has the value 0 and the lines have the value 1. This information is used in the later image process stage.

4.2.2 Line Extraction

It is not good practice to attempt to find corresponding points for all pixels in the projected lines between the two images [55]. The projected lines usually need to be thinned to one-pixel-thick lines to increase the accuracy of the matching between the images.

A skeletonizing method [25] is implemented to thin the projected lines. The skeletonizing, also referred to as line thinning, shrinks each projected line by repeatedly deleting edge points. The thinned lines are approximately at the center of the original projected lines. The algorithm operates under three basic constraints: 1) it does not

Figure 4.8: Projected line image after local thresholding.

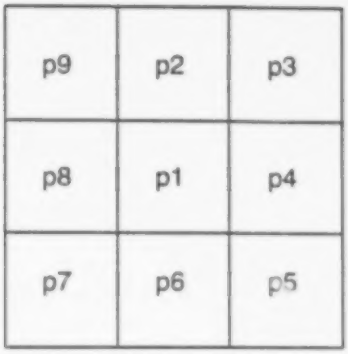

Figure 4.9: Neighbourhood arrangement used by the thinning algorithm.

remove end points. 2) it does not break the lines, and 3) it does not cause excessive erosion of the region.

The implementation of this method for images consists of two basic steps. First, a point p1 (see Figure 4.9) is flagged for deletion if the following conditions are satisfied.

1) $2 \leq N(p_1) \leq 6$,

2) $S(p_1) = 1$,

3) At least one of (p_2, p_4, p_6) is zero,

4) At least one of (p_4, p_6, p_8) is zero.

Where $N(p_1)$ is the number of nonzero neighbours of p_1, which is

$$N(p_1) = p_2 + p_3 + \cdots + p_8 + p_9 \tag{4.6}$$

and $S(p_1)$ is the number of 0-1 transitions in the ordered sequence of $p_2, p_3, \cdots, p_8, p_9, p_2$.

It can be seen from Figure 4.9, the first condition implies that p_1 is not a edge point, when $N(p_1)$ is 1 or is greater than 6. When $N(p_1)$ is 1, p_1 is the end point of the line and should not be deleted. When $N(p_1)$ is 7, p_1 is inside the region, and deleting it would cause erosion into the region. The second condition implies

that when $S(p_1)$ does not equal to 1, deleting p_1 would disconnect segments of the lines during the thinning operation. Satisfying the last two conditions simultaneously means that p_4 equals to 0 or p_6 equals to 0 or both p_2 and p_8 are equal to 0. A point that satisfies these conditions, as well as the first and second conditions, is an east or south boundary points or a northwest corner point in the boundary. In either case, p_1 is flagged to be removed.

If one or more conditions are violated, the value of p_1 is not changed. The point p_1 is flagged for deletion only when all the conditions are satisfied. After the first step has been applied to all the points in the image, points that are flagged are deleted (changed to 0). After all the edge pixels found in the first step are removed, the second step is applied to the resulting data in exactly the same manner. The point p1 is flagged for deletion if the following constraints are satisfied.

1) $2 \leq N(p_1) \leq 6$,

2) $S(p_1) = 1$,

3) At least one of (p_2, p_4, p_8) is zero,

4) At least one of (p_2, p_6, p_8) is zero.

First two conditions in the second step are the same as in the first step and only the last two conditions are changed. Satisfying the last two conditions simultaneously means that p_2 equals to 0 or p_8 equals to 0 or both p_4 equals to 0 and p_6 equals to 0. These correspond to north or west boundary points or a southeast corner point in the boundary and should be removed.

The thinning algorithm applies this basic two-step procedure iteratively until no further points are deleted at which time the algorithm terminates and the resulting lines are approximately at the center of the original projected lines.

Figure 4.10: Projected lines image after applying thinning algorithm.

Figure 4.10 shows the results of the projected lines after applying the thinning algorithm. Since only the edge pixels are removed during the thinning process, the resulting lines are approximately the center of the original projected lines. The resultant image contains broken lines due to different skin reflectance and depth modulations of the object surface, as shown in the figure. To alleviate this problem, further processing is needed.

4.3 Line Search

After the thinning process the projected line image often contains broken line segments due to different skin reflectance and depth modulations of the object surface. The line search is a procedure that is used to connect the broken line segments into continuous lines. There are three stages in the line searching procedure. Each stage reconnects different kinds of broken line segments. The first two stages use neighbourhood relations between adjacent pixels of the lines to form continuous lines. The third stage uses parallel properties and vertical spatial relationship of the lines to reconnect the lines.

4.3.1 Three Stages of Line Search Procedure

Stage One - Fills in the missing points of the lines

The first stage of the line search procedure is governed by the constraints that resulting lines are to be smooth and continuous. Based on neighbourhood relations between adjacent pixels of the lines, this part of line search procedure connects broken lines that are close to each other. In this stage, when a broken line segment of an end point is found, a search for another end of line segment point is performed horizontally in a 3 by 5 window. The missing data points are generated by using linear interpolation between the end points. It fills a maximum gap of 4 pixels. The 3 by 5 window is chosen arbitrarily. It is used to ensure that the closed broken line segments are reconnected first. Since the distance between the centers of the two projected lines is approximately 10 pixels, this local search does not falsely reconnect line segments that are not associated with each other. The first stage solves most of the broken line problems due to different skin reflectance. However, sudden changes of depth in the back surface, such as on a prominent scapula, could cause a large disruption of the line's connectivity. Additional stages are required to connect the remaining broken lines.

Stage Two - Connects broken line segments due to depth modulation

The second stage of line search procedure is based on the smoothness criterion, and it reconnects broken line segments based on the distance and orientation of the line segments. This stage starts with finding both start and end points of the lines. Since the broken line segments can be reconnected only between the start and the end points, the distances between a start point of a line and the end points of the remaining lines are calculated. If the distance is smaller than a threshold value of 10 pixels, these two line segments are considered as a candidate pair of broken lines that potentially might be reconnected. The threshold value of 10 pixels is chosen because it is found that most missing pixels due to depth modulation in the trunk surface are within 10 pixels. In addition, since the distance between the centers of the two projected lines is approximately 10 pixels, to avoid falsely reconnecting the lines, a threshold value of 10 pixels is selected. To join the lines, the inclination angle between the start and end points of the candidate lines is examinated. If the inclination angle is less than ± 45 degrees, the two line segments are joined. The inclination angle of ± 45 degrees ensures that the line segments to be joined have similar orientations, and the resultant connected lines are smooth and without folding and sudden jumps. This procedure is applied repeatly, stopping when no further broken line segments are found. This stage solves most of the broken line problems due to the sudden change of depth in the object surface. However, it can not solve the problems which occur when the object surface has a large area of sudden depth change. In this case, a sequence of neighbouring lines are broken. The smoothness constraint is insufficient to decide where a line should be connected. Additional constraints such as parallel properties (contours can never cross) and vertical spatial relationship of the lines are needed.

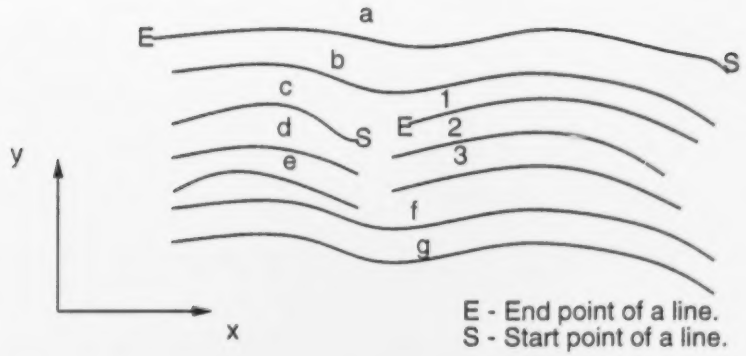

Figure 4.11: An example of sequence of neighbouring broken lines.

Stage Three - Connects series of neighbouring broken lines

The third stage is developed based on the constraints that the projected lines can never cross each other and are contiguous. This stage is designed to reconnect a series of neighbouring broken lines into continuous lines. These broken lines usually are caused by sudden depth change in the object surface. Figure 4.11 shows the problem of neighbouring broken lines. To solve this problem, a two-step procedure is implemented. The first step is similar to stage two, it starts with a search of the candidate of the broken lines that potentially may be reconnected. Broken lines are candidates for reconnection if the distance between the start point of a line and the end points of the rest of the lines is less than 30 pixels. The threshold value of 30 pixels is selected arbitrarily. It is used to ensure that only related broken line segments are selected. Based on this constraint, it is possible that line 'c' is connected to either lines '1' or '2', line 'd' to either line '2' or '3', and line 'e' to line '3'.

To decide where the connection should be made, the y coordinate of the parent broken line segments, eg. segment 'c', 'd' and 'e', are calculated. The y coordinate of the parent segments are obtained by calculating the average of y coordinates of all the pixels of the line segments. After sorting the y coordinate values, the parent segment with the largest y coordinate is selected (segment 'c'). Similarly, the y coordinate

of the mature lines, eg, line 'a', 'b', 'f', and 'g', are calculated. The mature line with shortest distance to the the parent segment (segment 'c') found previously is selected (line 'b'). Lastly, the y coordinate of the candidates of the broken segments (segment, '1' and '2') of the parent segment found previously are computed. The candidate segment (segment '1') with the shortest distance to the mature line (line 'b') found previously is obtained.

Based on the constraints that projected lines are contiguous and never cross each other, the candidate segment (segment '1') are connected with its parent segment (segment 'c'). This two-step procedure is applied repeatly until no further parent broken segments are found. This stage solves a sequence of neighbouring broken lines due to sudden depth change in the surface. This stage, along with the previous two stages, generally can reconnect all the broken lines that occur in the projecting line images. However, if the projected lines are dense when projecting onto the object surface, incorrectly connected lines may occur during the line search procedure; thus, generating errors for the later image processing. In this case, there is a trade off between the validity and the density of the lines.

The line search procedure does not dissect the false interconnected lines and alter the incorrectly connected lines for the projected line images. Figure 4.12 (a) shows an original image of a step pattern object. The size of this step pattern object is 152 x 305 millimetres. The sudden change of depth in the two steps are 51 and 25 millimetres, respectively. Figure 4.12 (b) shows the image of the step pattern object with projected lines on it. Figure 4.12 (c) shows the result after the line search procedure. The resulting lines indicate that the line search procedure does not correctly connect the broken line segments into the continuous lines. When there were false interconnected lines in the projected line image, the line search procedure did not dissect them, and alter the incorrectly connected lines. The line search procedure

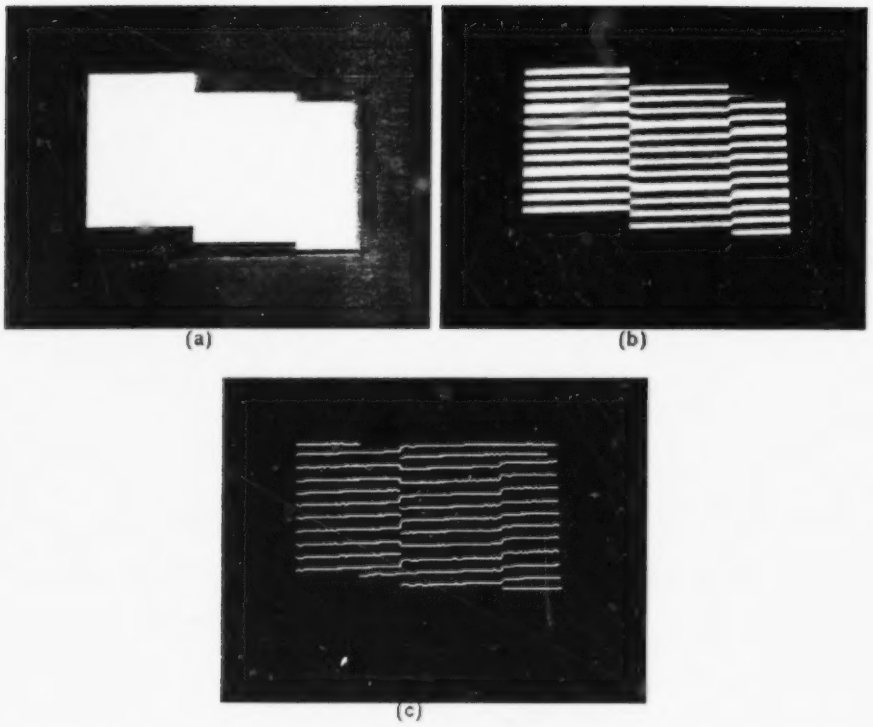

Figure 4.12: Sequence of processing steps for the step pattern object.

fails in this particular case. Human trunk surfaces are smooth and continuous. These sudden disruptions in the surface seldom occur and they are not a significant concern for this research.

4.3.2 Line Reconnecting Method

A midpoint line drawing algorithm [19] is used to connect large line disruption in the images. The algorithm computes the coordinates of the pixels that lie on or near an ideal straight line. This algorithm uses only integer arithmetic, and it allows the calculation of the next pixel to be performed incrementally by using the calculation already done at the current pixel. Consider the line in figure 4.13, where the previously selected pixel is indicated by a black circle and the two unfilled circles indicate the

pixels from which the next stage is to be chosen. Assume that the selected pixel P is at (x_p, y_p) and the next two pixels are E and NE. Let Q be the intersection point between the line and the grid line x = x_p + 1, and M be the midpoint between pixels E and NE. It can be seen from the figure when the midpoint lies above the line, pixel E is closer to the ideal line. When the midpoint lies below the line, pixel NE is closer to the ideal line, and when the midpoint lies on the line, either pixel N or pixel NE can be selected. Based on these constraints, the midpoint algorithm chooses E as the next pixel for the line shown in figure 4.13.

To decide on which side of the line the midpoint lies, a decision variable d, a function of (x_p + 1, y_p + 1/2), is computed. Since the form of the line equation can be written as:

$$F(x, y) = dy \cdot x - dx \cdot y + B \cdot dx = 0 \tag{4.7}$$

where $dy = y_n$ - y_0, $dx = x_n$ - x_0 and (x_0, y_0), (x_n, y_n) are the endpoints on the lines. Coefficient B is a constant which cancels out in the final equation. The decision variable of the algorithm can be written as: d = F(x_p + 1, y_p + 1/2). It can be verified that if $d > 0$, the midpoint is below the line; if $d < 0$, the midpoint is above the line; and if $d = 0$, the midpoint is on the line and either pixel can be chosen. In this case, the pixel below the line is picked as the next point on the line.

Since the first pixel is the first endpoint (x_0, y_0), the initial value d calculated for choosing between E and NE is written as:

$$\begin{aligned} d &= F(x_0 + 1, y_0 + 1/2) \\ &= dy \cdot (x_0 + 1) - dx \cdot (y_0 + 1/2) + B \cdot dx \\ &= F(x_0, y_0) + dy - dx/2 \end{aligned} \tag{4.8}$$

But(x_0, y_0) is a point on the line and hence F(x_0, y_0) = 0; thus, d_{start} is just dy - dx/2. Based on the sign of d_{start}, the E or NE pixel is chosen.

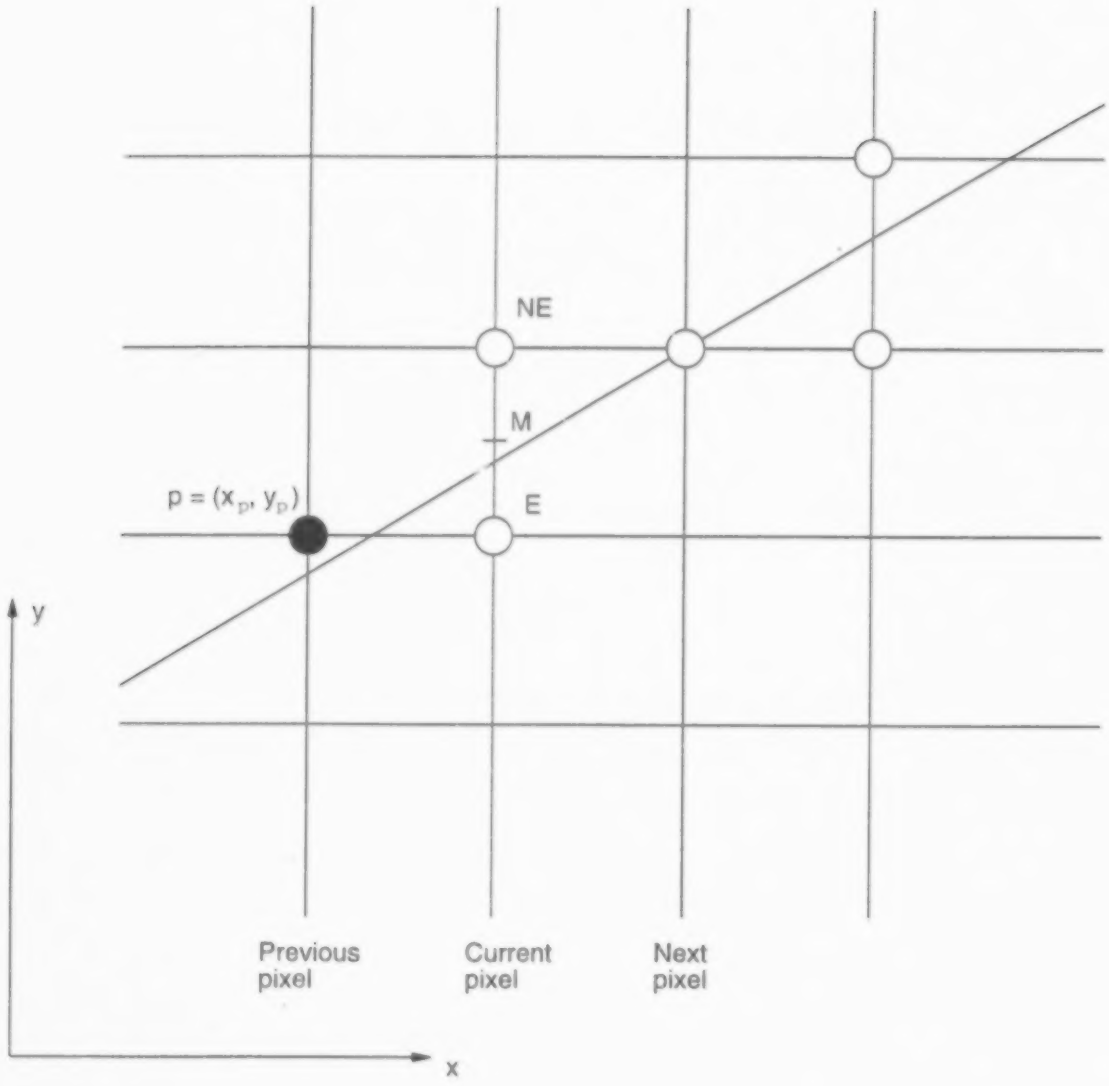

Figure 4.13: The pixel grid for the midpoint line drawing algorithm.

If pixel E is chosen, the next new decision variable is computed as:

$$\begin{aligned} d_{new} &= F(x_p + 2, y_p + 1/2) \\ &= dy \cdot (x_0 + 2) - dx \cdot (y_0 + 1/2) + B \cdot dx \end{aligned} \tag{4.9}$$

subtracting d_{start} from d_{new}, a incremental difference of dy is obtained. Thus, a increment is added to d after E is chosen: deltaE = dy.

Similarly, if NE is chosen, the next new decision variable is computed as follow,

$$\begin{aligned} d_{new} &= F(x_0 + 2, y_0 + 3/2) \\ &= dy \cdot (x_0 + 2) - dx \cdot (y_0 + 3/2) + B \cdot dx \end{aligned} \tag{4.10}$$

subtracting d_{start} from d_{new}, the incremental difference of dy-dx is found. Thus, the increment is added to d after NE is chosen: deltaNE = dy - dx. The value of the decision variable at the next step is incrementally computed by adding either deltaE or deltaNE to the current calculated value.

The implementation of the midpoint algorithm consists of three steps: First, the initial decision variable d_{start} is computed. Second, the next pixel is chosen based on the sign of the decision variable calculated in the previous step. Third, the decision variable is incremented by adding either deltaE or deltaNE to the current value, The next pixel is selected based on the sign of the new decision variable. This process is repeated until the broken lines are joined at which time the algorithm terminates.

4.4 Line Sequence Analysis

Line sequence analysis is a process for identifying and numbering the continuous lines in the image. Eight-connectivity [18] is used. Eight-connectivity classifies each continuous line(connected components) only once, and prevents conflict of line identification. To label and identify the lines in the image, the image is scanned pixel by pixel, from left to right and from top to bottom.

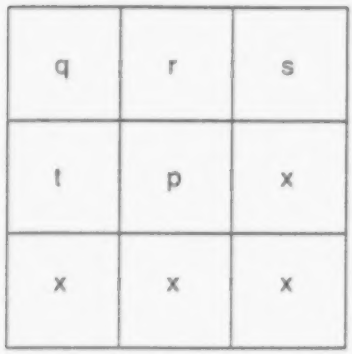

Figure 4.14: Neighbourhood arrangement used by the eight-connectivity.

As shown in figure 4.14, if p is the pixel at any step in the scanning process and r, and t are the upper and left-hand neighbours of p, respectively and q and s are the two upper diagonal neighbours of p, the following procedure is implemented. If the value of p is 0, moves on to the next scanning position. If the value of p is 1, pixels q, r, s and t are examined. If pixel q, r, s and t are all 0, a new label is assigned to p, because this is the first time that this connected component has been encountered. If only one of the four neighbours is 1, the label of that pixel is assigned to p. If two or more neighbours are 1, and had the same label, the label of one of the pixels is assigned to p. If two and more neighbours are 1, but have different labels, the label of one of the neighbour pixel is assigned to p and a note that two neighbour pixels are equivalent is made. At the end of the scanning process, all points with value 1 are labelled, but some of the labels are equivalent. To convert the equivalent labels to the same label, the equivalent label pairs are sorted into equivalence classes, and a unique label is assigned to each class. Then a second pass through the image replaces each label by the label assign to its equivalence class. At the end of the eight-connectivity procedure, each line in the image has a unique label assigned to it.

Chapter 5

THREE-DIMENSIONAL DATA ACQUISITION

5.1 Line Correspondence Between Stereo Image Pair

After the projected lines are numbered, determining line correspondence between stereo pair images is the next stage of image processing. In this process, the thick lines are first identified based on their relative thickness and brightness. The thick lines between the top and the bottom camera images are then matched based on the line sequence. To match the rest of the lines between the images, constraints such as the order of the lines and relative location of the thin and thick lines are used. There are two stages for achieving line correspondence: identification of the thick lines and correlation of the thin lines.

Identification of The Thick Lines

The thick lines in the line images are identified based on the relative thickness and brightness. There are three steps to find the thick lines in the images. First, the total and the mean intensity value of the cross section of the projected lines is calculated. The projected line with the maximum total intensity value is selected and considered to be one of the thick lines. This step takes the line with the highest total intensity value as a thick line. This ensures the correctness of the thick line identification. Second, after one of the thick lines is identified, to find the remaining thick lines,

the absolute differences in mean intensity value between the selected thick line and the remaining projected lines are calculated. If the difference is less than a threshold value of 10, the lines are considered to be thick line candidates. The threshold value of 10 was chosen because it was found that when thick lines are projected on the trunk surfaces, they had similar mean intensity values, and their mean intensity values were larger than the mean intensity values of the thin lines by 10. The threshold value of 10 is the preliminary constraint to separate the thick lines from the thin lines. To correctly identify the thick lines from the candidates, in the third step, the precoded line pattern relationship between the thin and the thick lines described in section 3.2 is used. The code is obtained by arranging the thin lines in groups and the thick lines located at 0, ± 5, ± 10, ± 15 ... . Based on this line code, the number of the thin lines between the the candidates of the thick lines and the selected thick line is examined. If the number of the thin lines between the selected thick line and the thick line candidates satisfies the precoded line pattern constraint, the candidate lines are considered to be thick lines. The relative brightness of the lines and the relative line location of the thick lines with respect to each other ensure that only the thick lines in the line projected image are identified.

Because the two cameras are positioned to have approximately the same field of view as described in section 3.1, there are the same number of corresponding thick lines in both of the stereo camera images. Therefore, once the thick lines are identified in the two images, the thick line correspondence is based on the relative image y coordinate of the thick lines. A thick line with the maximum y coordinate in one image matches with a thick line with the maximum y coordinate in another image. A thick line with the second largest y coordinate in one image matches a thick line with the second largest y in another image. This matching procedure is continued until all the thick lines in both images are correlated.

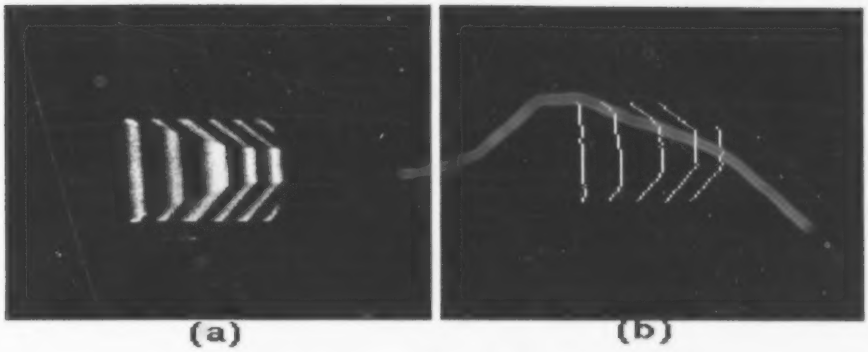

Figure 5.1: Sequence of processing steps for the pyramid shaped object.

This thick line matching scheme has a limitation when being used on a small object. This is because this scheme requires thick line correspondence to be determined first. If the thick lines are missing from the projected line images, line correspondence between two images can not proceed. Figure 5.1 illustrates this problem. Figure 5.1 (a) shows the original image of an pyramid shaped object. The size of the pyramid base is 76 by 76 millimetres. Figure 5.1 (b) shows the result after the feature extraction process. However, due to the small size of the object, the thick projected lines which are used to make the initial matches between the stereo pair are missing from the original images; thus, the line correspondence between the two images fails. The matched points between the stereo pair are not found. The system does not yield the three-dimensional information of the object surface.

Correlation of The Thin Lines

To correlate the remaining thin lines in the two images after the thick lines are matched properly, constraints such as line order and relative location of the thin and thick lines are used. This stage correlates the thin lines by using the initial thick line matches. It starts at one of the thick line matches and searches upward and downward in both images finding the corresponding thin lines. If the order of two lines is the same in both images during the search procedure, these two lines are

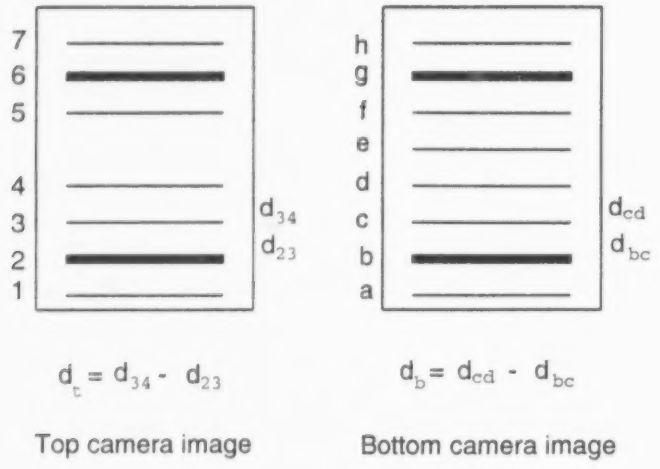

Figure 5.2: Line correspondence between the stereo pair.

considered to be matched. The search process terminates when the difference of the distance between three consecutive lines in either images is greater than a threshold value of 5 pixels, or when all the projected lines are matched. The threshold value of 5 was chosen because the pattern projected onto the trunk surfaces are horizontal with equally spaced parallel lines. The distance between two consecutive projected lines is similar. It was found that the difference of distance between three consecutive lines is within 5 pixels unless there are missing lines in the images in which case the line matching process terminates. This stage of searching the matched lines is applied repeatly, stopping when all the initial thick line matches are used.

Figure 5.2 shows an example of line correspondence between two images. In the figure, line '2' and 'b' and line '6' and 'g' are assumed to be the initial thick line matches between the top and the bottom camera images. If the searching procedure starts at line pair '2' and 'b', the searching procedure yields a matched line set contained with lines '1','2', '3' and '4' and lines 'a','b', 'c' and 'd'. The reason it did not find all the line matches is because the searching procedure is terminated when it found the difference between distance d_{54} and distance d_{43} is greater than a predefined threshold value. When the next initial match is used, for the same reason,

the searching procedure yields only lines '5', '6' and '7' in the top camera image matching with lines 'f', 'g' and 'h' in the bottom camera image, respectively. Thus, in this particular case, the combination of both sets of the matched line pairs are the results of the line correlation process. These results show that a missing line in the top camera image does not cause false matches between the images. All lines in both images are correctly matched. However, the line matching process does not yield correct matches when there is sudden large depth change in the horizontal direction in the object surface. A sudden large change in depth in the object surface may cause occlusion to occur in the stereo image pairs. Figure 5.3(a) and (b) show an example of occlusion in the stereo image pairs. Figure 5.3 (c) shows the image of (b) after the line search procedure. In this case, two thick lines on the cylindrical shaped object were initially matched. The remaining thick lines were not found the matches between the two images because the occlusion in the images caused the remaining thick lines to violate the precoded line pattern constraint. Satisfying the precoded line pattern constraint is one of the requirements to correspond the thick lines between the images. The matching process found the correspondence between the two images for the thin lines on the cylindrical object based on the two initial thick line matches. The remaining corresponding thin lines between the two images were not found because some of the thin lines were missed when occlusion occurred. The constraint of matching the thin lines was violated. As a result, only lines on the cylindrical object were correlated between the two images. Figure 5.3 (d) shows the image after the matching process. The line matching process produces matching results without knowing that occlusion occurred. However, trunk surfaces are smooth and continuous and these sudden disruptions in the surface seldom occur.

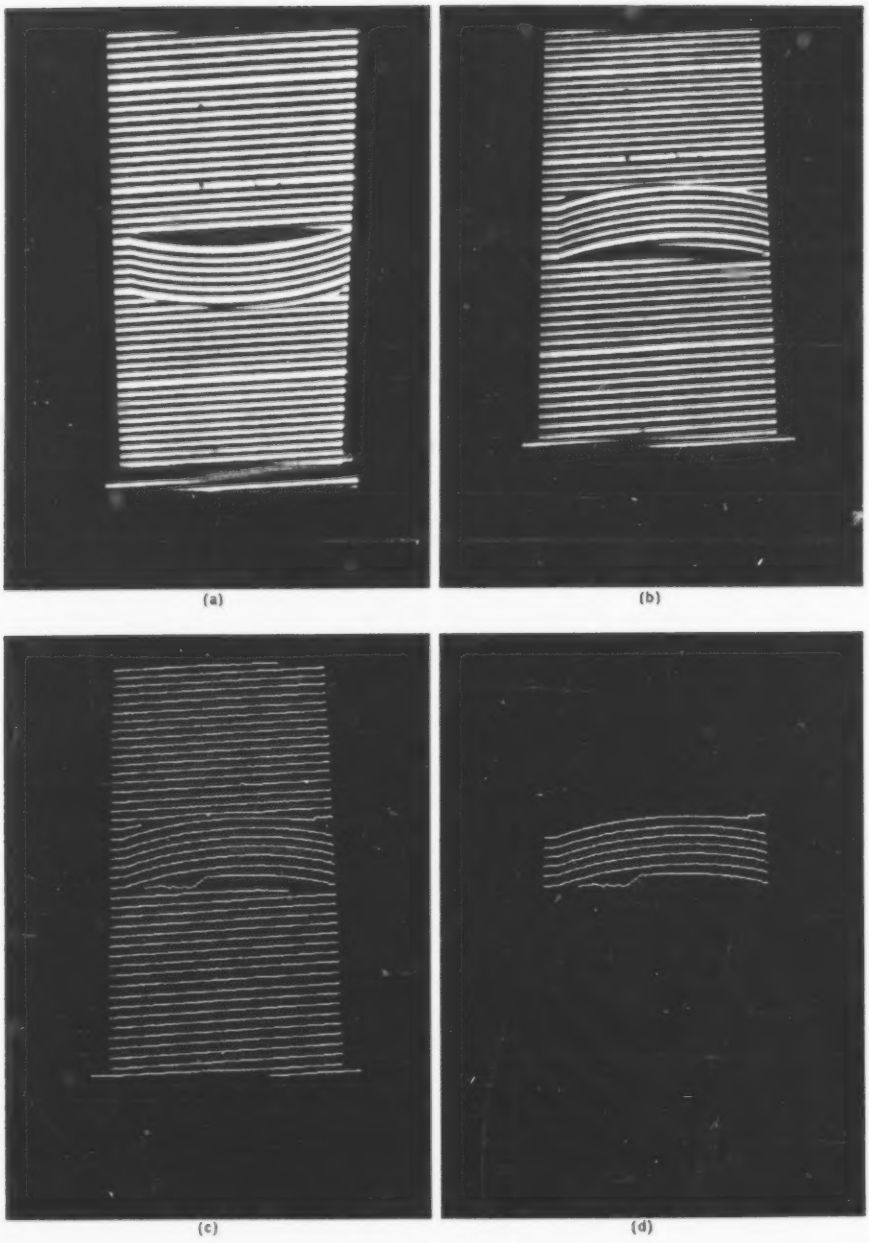

Figure 5.3: Occlusion in the stereo image pair.

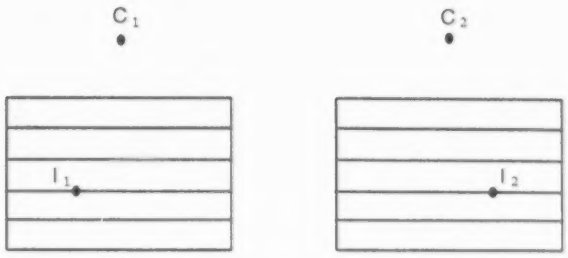

Figure 5.4: Parallel epipolar lines in both images.

5.2 Image Rectification

Correctly identifying the line pattern provides a one dimensional match between the top and the bottom camera images. To match the second dimension, the epipolar constraint is used. The epipolar constraint states that the corresponding points in a stereo image pair are located on a pair of epipolar lines. In a very special case shown in figure 5.4, when the line C_1C_2 is parallel to the image planes of both cameras, the epipolar lines are the scan lines of the stereo pair parallel to the axes of the image coordinate frame(epipolar geometry). However, this situation is very difficult to obtain in practice as the two cameras are usually not parallel to the line (C_1C_2). A process of transforming the camera image planes into planes parallel to the object frame is needed. This image transformation process is often called image rectification.

5.2.1 Background of Image Rectification

Image rectification is the process of making an equivalent vertical image from a tilted image [3]. After image rectification, the resulting image pair is recorded in epipolar geometry. The scan lines of stereo pairs are epipolar lines parallel to the axes of the image coordinate frames. Hence, only a one dimensional match is required between the stereo pairs.

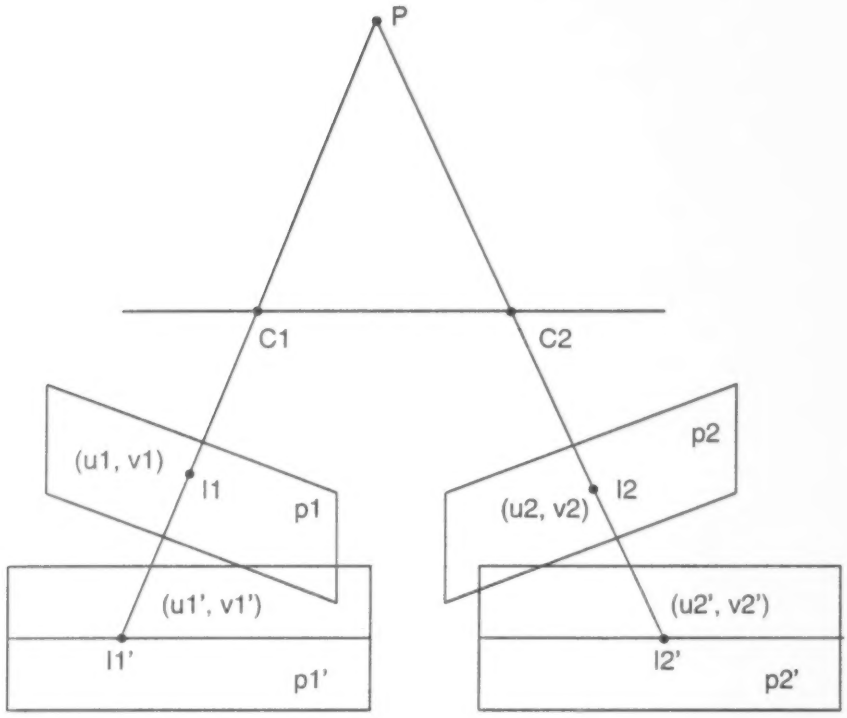

Figure 5.5: Geometry of image rectification.

Figure 5.5 shows the basic geometry of image rectification. The objective of image rectification is to define two new transformation matrices, so that the rectified images have the same optical centers C_1 and C_2, respectively but with a new common image plane $p1'$ and $p2'$ parallel to line C_1C_2. Rectification computes new coordinates ($u1'$, $v1'$) and ($u2'$, $v2'$) from the original ones ($u1$, $v1$) and ($u2$, $v2$).

There are many ways to compute the new transformation matrices. Ayache and Hansen [4] computed the transformation matrices based on constraints of the parallel epipolar lines. Their approach did not require the computation of the extrinsic parameters and the intrinsic parameters of the cameras Their method used a least squares procedure to estimate the transformation matrices and involved complicated mathematics.

Cho and Schenk [11] described a procedure of computing rectified images of aerial photographs with respect to the object space. In their method, the transformation matrices of the rectified images were computed based on the exterior orientation of the stereo pair. This technique required cameras to be first calibrated and the images were required to be adjusted so that they were free of systematic distortion. There were two steps involved in the transformation of the pixel images to rectified images. First, pixel images were transformed to vertical images. This involved an orthogonal rotation matrix, **M**, defined by the orientation of the cameras. Next, a transformation from vertical images to rectified images was applied. This transformation was computed based on perspective centers and exterior orientation angles of the cameras.

5.2.2 Implementation Of Image Rectification

The image rectification method developed by Cho and Schenk was used in this research because after the camera calibration procedure the cameras' exterior and interior parameters are known. The first stage of image rectification is to compute the orthogonal rotation matrix which transforms the pixel image pair into the vertical image pair. The orthogonal rotation matrix [56], **M**, is defined by the exterior orientation angles of the camera. It is written as:

$$\mathbf{M} = \begin{bmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & m_{33} \end{bmatrix} \tag{5.1}$$

where

$$m_{11} = \cos\phi\cos\kappa$$

$$m_{12} = \sin\omega\sin\phi\cos\kappa + \cos\omega\sin\kappa$$

$$m_{13} = -\cos\omega\sin\phi\cos\kappa + \sin\omega\sin\kappa$$

$$m_{21} = -\cos\phi\sin\kappa$$

$$m_{22} = -\sin\omega\sin\phi\sin\kappa + \cos\omega\cos\kappa$$

$$m_{23} = \cos\omega\sin\phi\sin\kappa + \sin\omega\cos\kappa$$

$$m_{31} = \sin\phi$$

$$m_{32} = -\sin\omega\cos\phi$$

$$m_{33} = \cos\omega\cos\phi$$

and where ω, ϕ and κ are the rotation angles of the camera about X-axis, Y-axis and Z-axis, respectively.

The second stage involves a transformation from the vertical image to the normalized rectified image. The angles of the this transformation matrix R_B are K about the Z-axis, Φ about the Y-axis, and Ω about the X-axis. These angles are a function of the perspective centers and are expressed as:

$$K = \tan^{-1}\frac{BY}{BX}$$

$$\Phi = -\tan^{-1}\frac{BZ}{(BX^2 + BY^2)^{1/2}} \tag{5.2}$$

$$\Omega = \frac{\omega_1 + \omega_2}{2}$$

where BX = X_2 - X_1, BY = Y_2 - Y_1, and BZ = Z_2 - Z_1. The X_1, Y_1, Z_1, X_2, Y_2, and Z_2 are the coordinates of the perspective centers of the cameras.

The rotation matrix R_B is defined as:

$$R_B = R_\Omega R_\Phi R_K \tag{5.3}$$

where

$$R_K = \begin{bmatrix} \cos K & \sin K & 0 \\ -\sin K & \cos K & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$R_\Phi = \begin{bmatrix} \cos\Phi & 0 & -\sin\Phi \\ 0 & 1 & 0 \\ \sin\Phi & 0 & \cos\Phi \end{bmatrix}$$

$$R_\Omega = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\Omega & \sin\Omega \\ 0 & -\sin\Omega & \cos\Omega \end{bmatrix}$$

The new transformation matrix R_N which transforms the pixel images to the normalized rectified image is a multiplication of two rotation matrices. It is expressed as:

$$R_N = R_B \mathbf{M^T} \tag{5.4}$$

Since both rotation matrices $\mathbf{M^T}$ and R_B are known, the new transformation matrix R_N is computed for each of the image in the stereo pair by using Eq. 5.4. The collinearity equations are then used for the transformation of the pixel image to a rectified image. The transformation is represented in the following equation.

$$x_N = -f\frac{r_{11}x + r_{12}y - r_{13}f}{r_{31}x + r_{32}y - r_{33}f} \tag{5.5}$$

$$y_N = -f\frac{r_{21}x + r_{22}y - r_{23}f}{r_{31}x + r_{32}y - r_{33}f}$$

where r_{ij} are the elements of the rotation matrix R_N, f is the focal length of the camera, x and y are the image coordinates and x_N and y_N are the rectified image coordinates.

After both pixel images are rectified, the following procedures are used to define the rectified image coordinate system: 1) Determine the minimum of x and y coordinates of both rectified images. These values define the origin (row 0, column 0) of both rectified images. The rest of the pixels in the rectified images are translated, so that all the pixel coordinates of the rectified images are positive values. 2) Determine

the maximum of x and y coordinates of both rectified images and, using the minimum and the maximum of x and y coordinates, compute the distance in both the x and y direction for the rectified images. The maximum distance of the images in both x and y direction determines the size of the epipolar pixel image in the rectified image coordinate system.

After rectifying both the top and bottom camera images and transforming them into the new image coordinate system, the scan lines of the rectified stereo pair were epipolar lines parallel to the axes of the image coordinate frames. Hence, the potential matched points along the matched lines between the two images were found by searching vertically along the scan lines of the images

5.3 Point Correspondence

Once line patterns between two images are matched, and the images are rectified, finding the correlation of the points between the stereo images becomes straight forward. According to the epipolar constraint, the potentially matched points between two images are on the same epipolar line. Since the stereo image pair has been rectified, the epipolar lines are the scan lines of the images. Corresponding points between two images are found by searching points that are on the same scan line along the matched lines in two images, In other words, the corresponding points between two images are the corresponding intersection points of the matched lines and the scan lines.

5.4 Three-Dimensional Surface Reconstruction

Calculation of three-dimensional data is straight forward once correspondence between the stereo image pair is achieved. As described in section 2.5, there are many approaches that can be used to reconstruct three-dimensional surface information.

The collinearity equations were used in this research.

5.4.1 Computation of 3D Data Using Collinearity Equations

The collinearity equations described in section 2.5 are based on the assumption that an object point, its image point, and the perspective center of the camera, lie on a straight line. As a result, the collinearity equation(Eq. 2.1) is written as:

$$\bar{\alpha} = \kappa \mathbf{MA} \tag{5.6}$$

where $\bar{\alpha}$ is a three dimensional image vector, κ is an unknown scale factor, and M is a 3-by-3 orthogonal transformation matrix. Parameter **A** is a vector from the perspective center to the object point. These equations are non-linear. By using Taylor's theorem, collinearity equations in linearized form are written for each matched point of one image as [56]:

$$v_x = b_{11}dX + b_{12}dY + b_{13}dZ + c_1 \tag{5.7}$$

$$v_y = b_{21}dX + b_{22}dY + b_{23}dZ + c_2 \tag{5.8}$$

where

$$b_{11} = (x/q)m_{31} + (f/q)m_{11}$$

$$b_{12} = (x/q)m_{32} + (f/q)m_{12}$$

$$b_{13} = (x/q)m_{33} + (f/q)m_{13}$$

$$b_{21} = (y/q)m_{31} + (f/q)m_{21}$$

$$b_{22} = (y/q)m_{32} + (f/q)m_{22}$$

$$b_{23} = (y/q)m_{33} + (f/q)m_{23}$$

$$c_1 = (qx + rf)/q$$

$$c_2 = (qy + sf)/q$$

and

$$r = m_{11}(X - X_o) + m_{12}(Y - Y_o) + m_{13}(Z - Z_o)$$

$$s = m_{21}(X - X_o) + m_{22}(Y - Y_o) + m_{23}(Z - Z_o)$$

$$q = m_{31}(X - X_o) + m_{32}(Y - Y_o) + m_{33}(Z - Z_o).$$

The m_{ij}'s are the elements of the orthogonal transformation matrix, x and y are image coordinates of the point after correction for the camera distortions, X_o, Y_o and Z_o are the coordinates of perspective center of the camera, and f is the focal length of the camera. The radial and de-centering corrections to x and y were computed by using Eq 2.7.

Since the r, s, q and c_{ij} and b_{ij} coefficients are functions of the six elements of the exterior orientation of the two cameras which are known after the camera calibration procedures[8], the only remaining unknowns in Eqs. 5.7 and 5.8 are dX, dY, and dZ. These are corrections to be applied to the initial approximations for the spatial coordinates X, Y, and Z for the object point. Since match points between the top and the bottom images were found, a set of four equations was obtained. The three unknowns dX, dY, and dZ were computed using a least squares solution. The solution of X, Y, and Z was found by an iterative process which terminated when magnitudes of the corrections became less than 0.0000001.

5.4.2 Calculation of Initial Approximation of X, Y and Z

To compute the initial approximations of X, Y and Z, triangulation [56] was used. Consider the similar triangles in figure 5.6, a set of linear equations is developed and expressed as:

$$\frac{x}{X - X_c} = \frac{y}{Y - Y_c} = \frac{-f}{Z_c - Z} \tag{5.9}$$

Reducing,

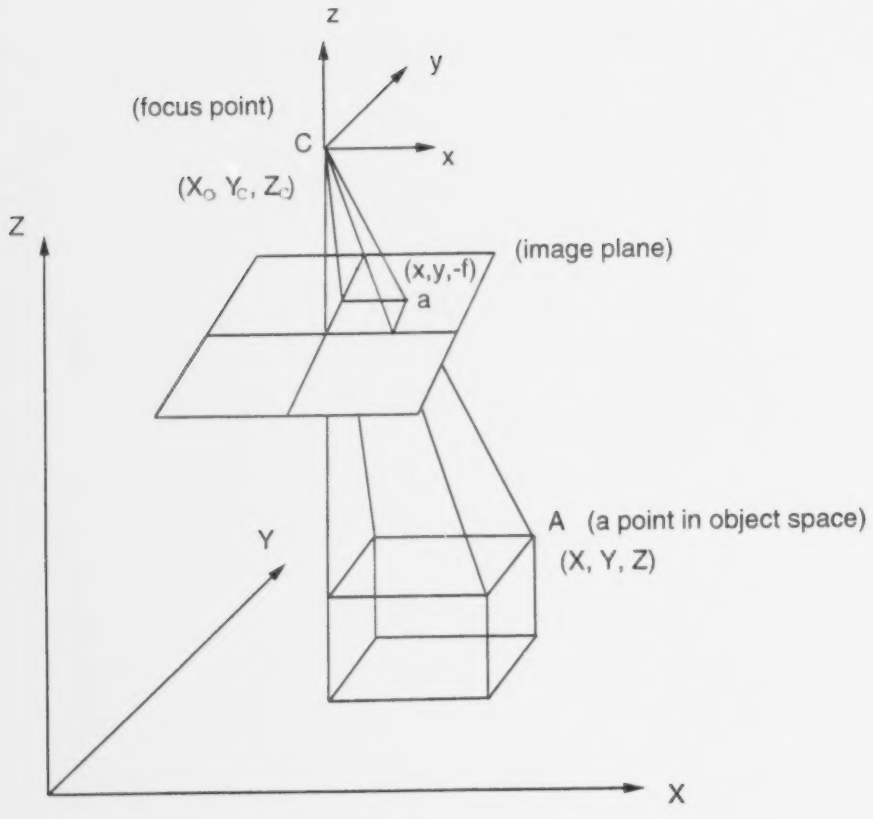

Figure 5.6: Geometry of 3D reconstruction.

$$X - X_c = \frac{Z - Z_c}{f} x$$

$$Y - Y_c = \frac{Z - Z_c}{f} y \tag{5.10}$$

$$Z - Z_c = \frac{Z - Z_c}{f} f$$

Replacing $\frac{Z-Z_c}{f}$ with λ_1, which is common to each of Eq. 5.10. The following expressions are written for the top camera image:

$$X = \lambda_1 x_1 + X_{c1}$$

$$Y = \lambda_1 y_1 + Y_{c1} \tag{5.11}$$

$$Z = \lambda_1 f_1 + Z_{c1}$$

Similarly the expressions for the bottom camera image are written as:

$$X = \lambda_2 x_2 + X_{c2}$$

$$Y = \lambda_2 y_2 + Y_{c2} \tag{5.12}$$

$$Z = \lambda_2 f_2 + Z_{c2}$$

using the first two equations of Eqs. 5.11, 5.12, λ_1 is:

$$\lambda_1 = \frac{y_2(X_{c2} - X_{c1}) - x_2(Y_{c2} - Y_{c1})}{x_1 y_2 - x_2 y_1} \tag{5.13}$$

Since the coordinates of the perspective centers (X_{c1}, Y_{c1}, Z_{c1}) and (X_{c2}, Y_{c2}, and Z_{c2}) and the focal length f_1 and f_2 of both cameras are known after camera calibration and the corresponding points between the top and the bottom camera images are found, image coordinates (x_1, y_1) and (x_2, y_2) of both images are known, using Eq. 5.13, λ_1 is calculated and then through Eq. 5.11 the initial approximations of X, Y, and Z are found.

5.4.3 Application of Least Squares Solution

After the pair of match points between the top and the bottom camera images are found, a set of four equations based on Eqs. 5.7 and 5.8 is obtained. The three unknowns dX, dY, and dZ are computed using the least squares solution [56]. The set of four equations are written as:

$$\mathbf{V} - \mathbf{C} = \mathbf{B}\Delta \tag{5.14}$$

where

$$\mathbf{V^T} = \begin{bmatrix} v_{1x} & v_{1y} & v_{2x} & v_{2y} \end{bmatrix}$$

$$\Delta^T = \begin{bmatrix} dX & dY & dZ \end{bmatrix}$$

$$\mathbf{C^T} = \begin{bmatrix} c_1 & c_2 & c_3 & c_4 \end{bmatrix}$$

$$\mathbf{B} = \begin{bmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \\ b_{41} & b_{42} & b_{43} \end{bmatrix}$$

Since matrix $\mathbf{C}$ is a constant matrix, taking partial derivatives with respect to matrix Δ of Eq. 5.14 yields following equation:

$$\frac{\partial \mathbf{V}}{\partial \Delta} = \mathbf{B}. \tag{5.15}$$

To arrive at the least square solution, the expression for the residuals $\mathbf{V^T V}$ is minimized. The partial derivative of $\mathbf{V V^T}$ with respect to Δ matrix is set to zero, yielding:

$$\frac{\partial \mathbf{V^T V}}{\partial \Delta} = \bar{0} = 2\mathbf{V^T}\frac{\partial \mathbf{V}}{\partial \Delta} \tag{5.16}$$

Substituting Eq. 5.15 into Eq. 5.16, produces:

$$2\mathbf{V}^{\mathbf{T}}\mathbf{B} = \bar{0} \tag{5.17}$$

and $\mathbf{B}^{\mathbf{T}}V$ is the transposed matrix of $\mathbf{V}^{\mathbf{T}}\mathbf{B}$; thus, it has:

$$\mathbf{B}^{\mathbf{T}}\mathbf{V} = \bar{0}. \tag{5.18}$$

By substituting Eq. 5.14 into Eq. 5.18, a normal equation for the Eq. 5.14 is obtained as follows:

$$\mathbf{B}^{\mathbf{T}}\mathbf{B}\Delta = \mathbf{B}^{\mathbf{T}}\mathbf{C}. \tag{5.19}$$

A basic least square matrix equation is obtained by multiplying both side of Eq. 5.19 by $(\mathbf{B}^{\mathbf{T}}\mathbf{B})^{-1}$ and reducing it, .

$$\Delta = (\mathbf{B}^{\mathbf{T}}\mathbf{B})^{-1}\mathbf{B}^{\mathbf{T}}\mathbf{C}. \tag{5.20}$$

Since the elements of matrices **B** and **C** are known from Eqs. 5.7 and 5.8, the matrix Δ containing with the corrections dX, dY and dZ can be calculated. The corrections are applied to the initial approximations of X, Y and Z. The revised values for X, Y, and Z are then obtained and the procedure is repeated. The solution of X, Y and Z for an object point is found when magnitudes of the corrections become less than 0.0000001.

To reconstruct the three-dimensional data for the surfaces, the X, Y and Z of each corresponding point found in the matching process described above is calculated. Figure 5.7 shows the three-dimensional information of the human trunk surface. The prominent features of the trunk surface are evidenced in this figure.

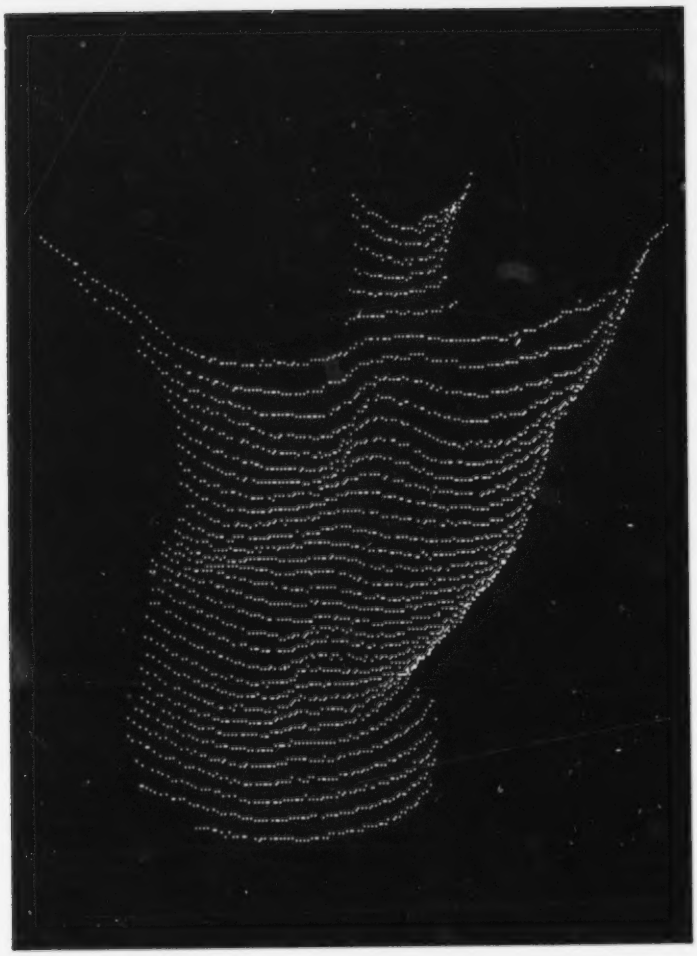

Figure 5.7: Reconstructed model of a human trunk.

Chapter 6

SYSTEM TESTING

The system was tested using three different shaped solid models to investigate the accuracy and reliability. The three different shaped solid models were: 1) a cylindrical object, 2) a step pattern object, and 3) a triangular shaped object. The system was then used to image scoliosis patients to confirm its practicality. The results show that the system has the potential of becoming a useful clinical tool.

6.1 System Testing Using Object Models

A cylindrical object was used to check the accuracy and reliability of the system. Figure 6.1 (a) shows an original image of this object. The width and the length of this object are 84 and 290 millimetres, respectively. The radius of the object is 223.8 millimetres. The object is used to model the shape of human trunk. Figure 6.1 (b) shows the image of the cylindrical object with the line pattern projected on it. To define the region of the cylindrical object in the projected line image, figure 6.1 (a) was used to define the boundary of the object. After the boundary of the cylindrical object was defined, the projected line image (figure 6.1 (b)) was segmented. Only the image data within the object boundary was considered for further processing. A local threshold window of 10 X 10 pixels was applied to the image to enhance the contrast of the projected lines. Figure 6.1 (c) shows the results of the line thinning algorithm

Trial	Comp. Radius(mm)	Standard Deviation	Error(mm)
1	225.3	0.63	+1.5
2	224.6	0.75	+0.8
3	226.3	0.69	+2.5
4	223.7	0.69	+0.1
5	225.1	0.57	+1.3
6	225.5	0.79	+1.7
7	222.5	0.51	-1.3
8	225.3	0.47	+1.5

Table 6.1: Accuracy of the eight trials of the reconstructed cylindrical object.

and line search procedure described in chapter 4. After line correspondence and image rectification, matched points in the stereo pair were found. The three-dimensional coordinate information of the cylindrical object was recovered using the collinearity equations. Figure 6.1 (d) shows the three-dimensional information of the cylindrical object. To test the accuracy and reliability of the acquired data, a total of eight trials placing the cylindrical object in different orientations and positions was conducted. Table 6.1 shows the accuracy of the reconstructed data of the eight trials after the data was fit using the least squares method fitting to a cylinder. The mean error of the radius of the eight trials in determining the radius of 223.8 mm cylindrical object was 1.4 millimetres. The largest error of the computed radius in the eight trials was 2.5 millimetres. These results demonstrate that the system is capable of accurately recovering reliable three-dimensional data for the cylindrical object.

Figure 6.2 shows the first trial of the Y and Z coordinate information of the reconstructed data in comparison with the measured circle of the cross section of the cylindrical object. In this trial, by using the maximum and the minimum of X and Y values, it was found the approximate width and the length of the reconstructed model of the cylindrical object were 79.9 and 284.2 millimetres, respectively. Comparing the computed width and length with the measured height and width of the cylindrical

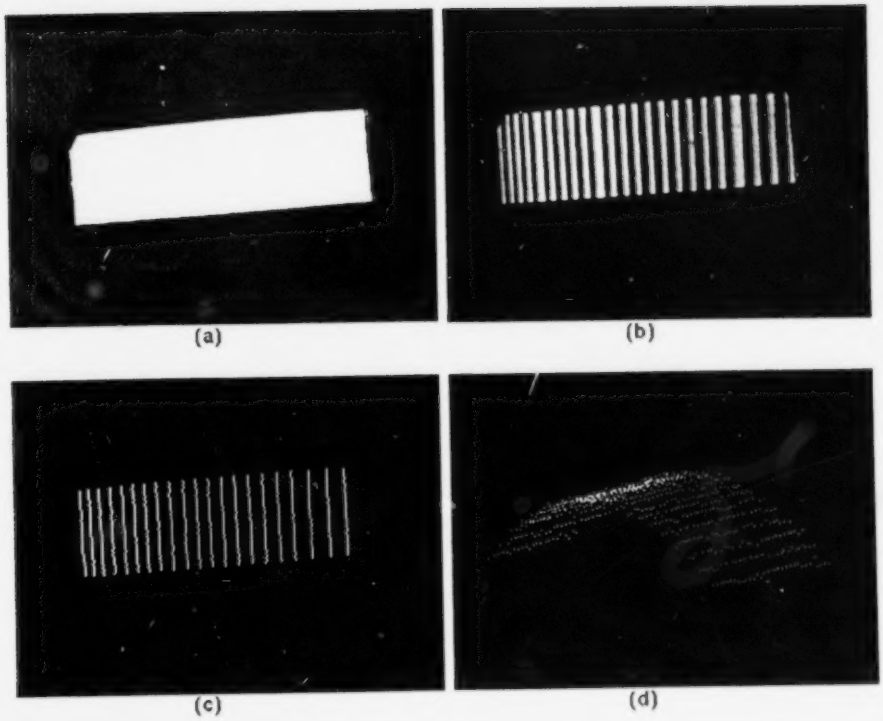

Figure 6.1: Sequence of processing steps for the cylindrical object.

object, approximate 95.1% of the height of the object and 98% of the width of the object were recovered.

To test the performance of the system when there were disruptions on the surface, a step pattern object was used. Figure 6.3 (a) shows the original image of one of the step pattern objects. The size of this step pattern object is 305 x 305 millimetres. The change of depth in the three steps are 10, 3, and 10 millimetres, respectively. Figure 6.3 (b) shows the image of the step pattern object with the lines projected onto it. Figure 6.2 (c) shows that although there are breaks in the projected lines, the line search procedure based on the neighbourhood relations between adjacent pixels of the lines successfully connected continuous lines. Figure 6.3 (d) shows the results after reconstruction of the step pattern object. In figure 6.3 (c), there are a total of 22 lines projected on the object. Since the object is 305 millimetres high, the average

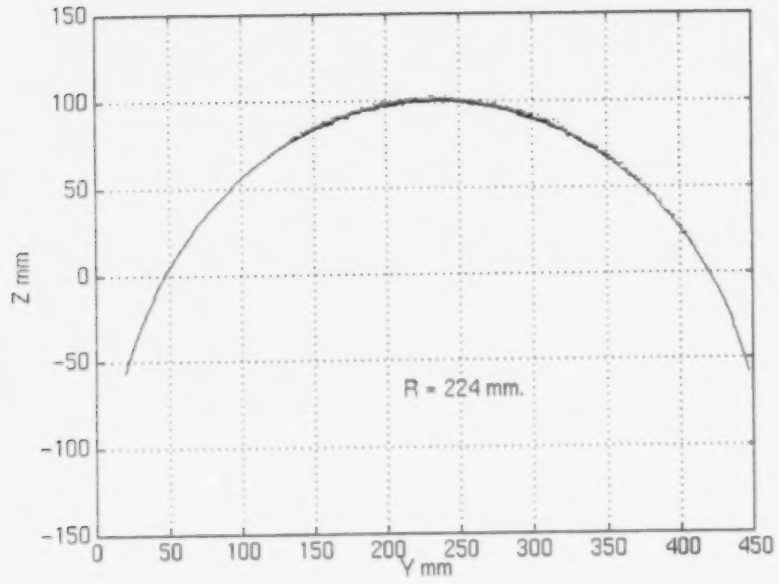

Figure 6.2: Y and Z coordinate information of the cylindrical object.

density of the projected line is approximately one line every 14 millimetres which means that the average density of the three-dimensional data in the vertical direction is approximately one line every 14 millimetres. In addition, it was found that the system yielded approximately 100 data points per line in this object. Since the width of the object is 305 millimetres, the average density of the three-dimensional data in the horizontal direction is approximately one point every 3 millimetres.

To test the accuracy of the acquired data, the least squares method of fitting the data to four planes was used after the reconstructed model of the step pattern object was approximately levelled. It was found that the transition data points in depth had large range of error when fitting to the planes, and the standard deviation of the overall fitting was ± 2.24. However, when the transition data points were not used, the range of error of fitting the planes was within ± 5 millimetres and the standard devi-

ation on the fit was 1.9 millimetres. By calculating the distances between the planes, the step changes were found 8.9, 2.6, and 8.6 millimetres, respectively. The error in reconstructed the depth changes were 1.1, 0.4, and 1.4 millimetres, respectively. Figure 6.4 shows the X and Y coordinate information of the step pattern object. By using the maximum and the minimum of the X and Y values, the approximate height and the width of the reconstructed model of the step pattern object were found 289 and 301 millimetres, respectively. Comparing the computed height and width with the measured height and width of the step pattern object, approximate 94.7% of the height of the object and 98.7% of the width of the object were recovered.

Figure 6.5 (a) shows an original image of a triangular shaped object with a notch in one of the inclined surfaces. This notch is used to simulate a skin fold which is common with children with significant scoliosis. The sharp changes in the triangular shaped object surfaces are used to simulate the scapular changes in the scoliosis patients. The inclination angles for the surfaces are 29° and 48°. The angle between the two inclined surfaces is 103°. The size of the notch in the inclined surface is 12 by 50 millimetres. Figure 6.5 (b) shows the image of a triangular shaped object with the line pattern projected on it. Figure 6.5 (c) shows the resultant image after the line thinning and search processes. Although there are sharp corners and gaps in the surfaces, the line linking process based on the spatial relationship between the projected lines correctly connected the broken lines to continuous lines. Figure 6.5 (d) shows the reconstructured triangular shaped object. The reconstruction of the object in area of the disruption is not accurate. This is because large disruptions of the lines are connected by using the line drawing algorithm. The line drawing algorithm computes the coordinates of the pixels that lie on or near a straight line defined by the end points. It connects the broken line segments with an interpolation of a straight line. Thus, it is possible that the lines drawn by the algorithm are not the same as the missed projected lines. However, since disrupted areas on back surfaces are generally small compared to the whole surface, the reliability of the three-dimensional data

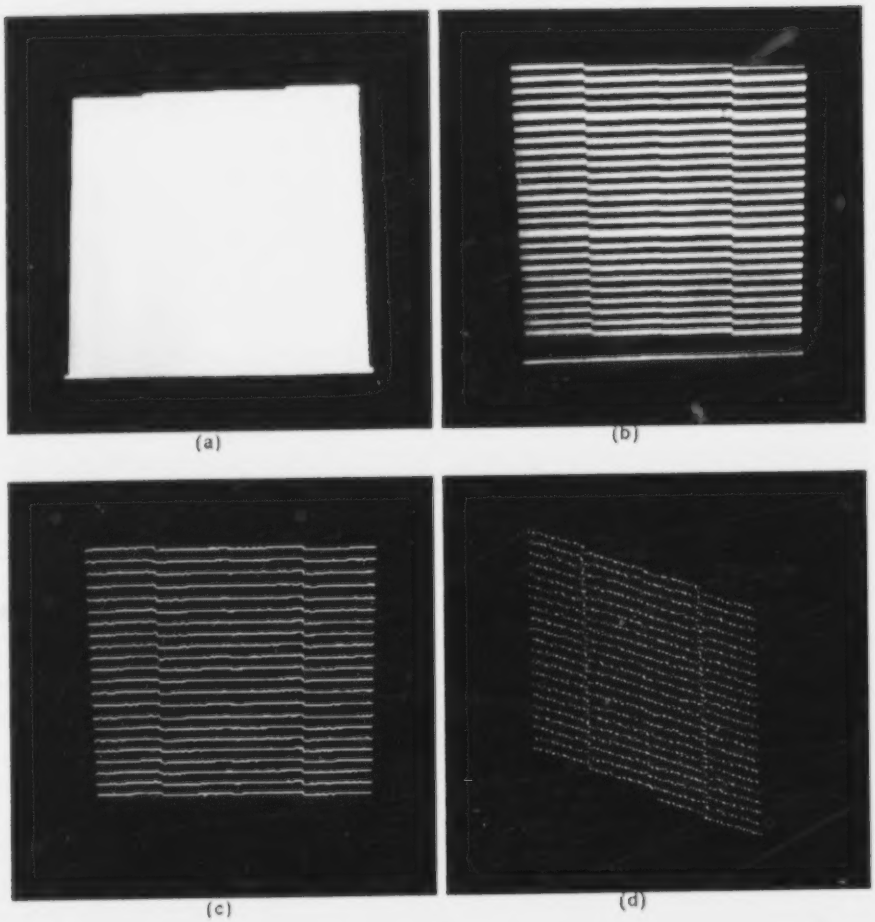

Figure 6.3: Sequence of processing steps for the step pattern object.

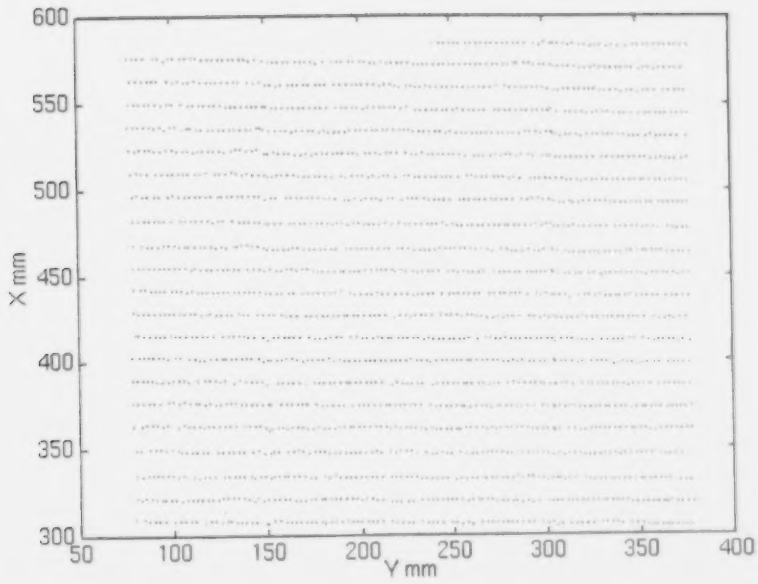

Figure 6.4: X and Y coordinate information of the step pattern object.

derived for the surface is not likely to be affected.

To test the accuracy of the acquired data, the least squares method of fitting the data to a straight line was used. Figure 6.6 shows the Y and Z coordinate information(cross section) of the reconstructed model of the triangular shaped object with their best fit lines. The line functions are $Z = 0.52Y - 36.214$ and $Z = -1.136Y + 460.900$ and their correlation coefficients(r^2) are 0.995 and 0.993, respectively. Using the slopes of the lines, the inclined angles of the lines were calculated. Table 6.2 shows the accuracy of the angles in the reconstructed model of the triangular shaped object.

The experimental results from the solid models indicate that the system is capable of accurately recovering reliable three dimensional coordinate information. The

Computed Angles(deg)	Known Angles(deg)	Error(deg)
27.5	29	1.5
48.6	48	0.6
103.9	103	0.9

Table 6.2: Accuracy of the angles of the reconstructed triangular shaped object.

experimental results from the triangular shaped object and the step pattern object indicate that although there are sudden changes of depth, sharp corners, and gaps in the surfaces, the image processing methods and algorithms can still recover the three-dimensional data for the surfaces. This system shows the potential of using in the reconstruction of the human trunks.

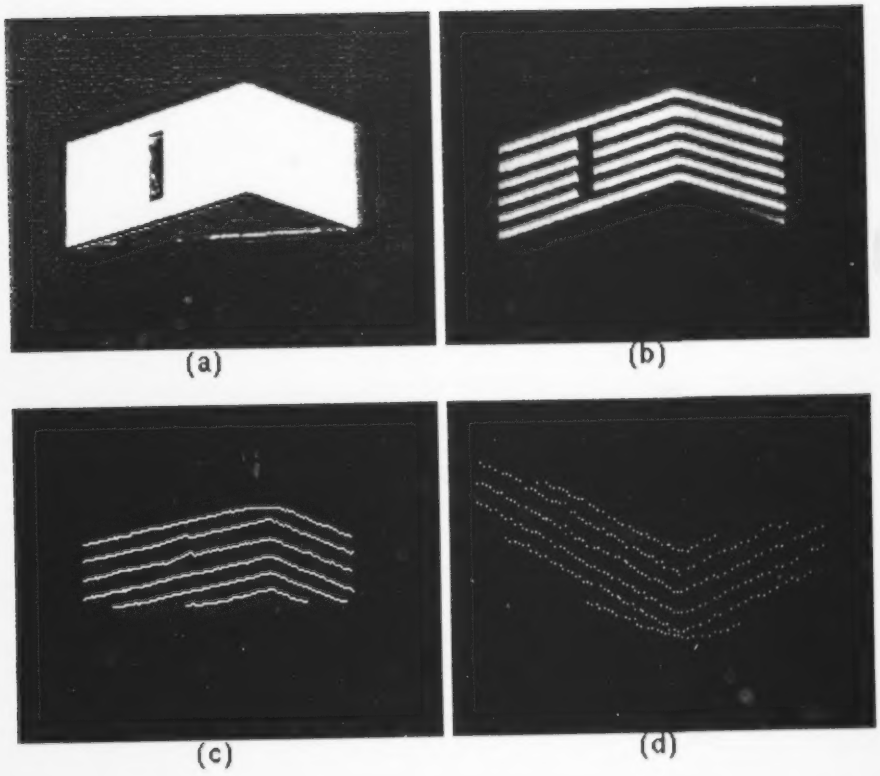

Figure 6.5: Sequence of processing steps for the triangular shaped object.

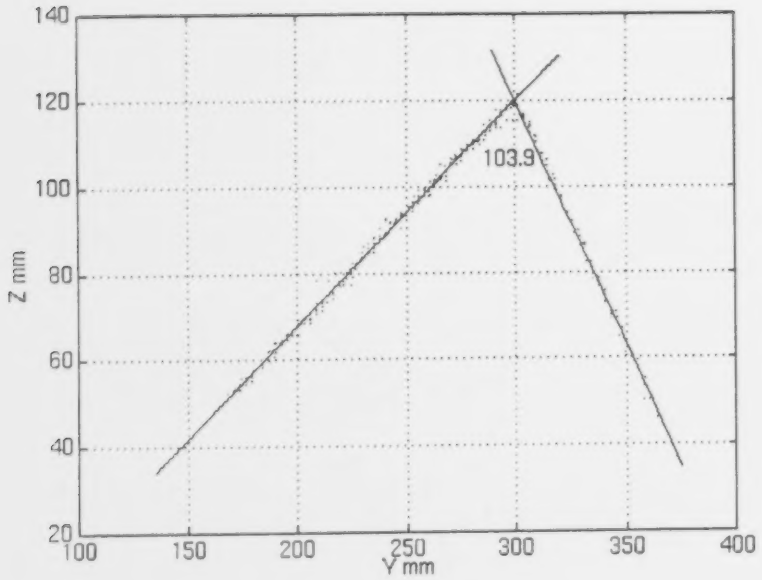

Figure 6.6: Y and Z coordinate information of the triangular shaped object.

6.2 Recovery of 3D Information from Scoliosis Patient Trunk Surfaces

The system was used to image scoliosis patients to confirm the practicality of the method. Figure 6.7 (a) shows an image of the trunk surface of a male test subject. This subject has mild scoliosis. His trunk is approximately 450 millimetres high. Figure 6.7 (b) shows an image of the patient trunk with the line pattern projected onto it. To define the region of the trunk for this image, figure 6.7 (a) was used. Region growing and edge detection methods were used to define the boundary of the trunk. After defining the boundary of the trunk, the projected line image (figure 6.7 (b)) was segmented. The image data within the trunk boundary were considered for further processing. A local threshold window of 10 X 10 pixels was applied to the image to enhance the contrast of the projected lines. Figure 6.7 (c) shows the results

of the line thinning algorithm and line search procedure. After line correspondence and image rectification, matched points between the stereo pairs were found. The three-dimensional coordinate information of the the trunk was recovered by using the collinearity equations. Figure 6.7 (d) shows the three-dimensional information of the trunk surfaces of the scoliosis patient. There are a total of 31 lines in the trunk that have been extracted from the projected line images. Since the height of the trunk is 450 millimetres, the average density of the three-dimensional data in the vertical direction is approximately one line every 14.5 millimetres. Similar features of the trunk surface can be seen between the two-dimensional image and the reconstructed points, figure 6.7 (a) and figure 6.7 (d).

Figure 6.8 (a) shows an image of the trunk surface of the female test subject with a light gown on her. Figure 6.8 (b) shows an image of the patient trunk with the line pattern projected on it. To define the region of the trunk for this image, Figure 6.8 (a) was used. Region growing and edge detection methods were used to define the boundary of the trunk. Figure 6.8 (c) shows the resultant image after edge detection. The boundary of the trunk is incorrectly defined. Instead of finding the boundary of the trunk, the boundary of the contrast region is found. This is because the contrast between the gown and the patient's trunk is low, and the region growing procedure fails to define the region of trunk as described in section 4.1. Figure 6.8 (d) shows the results of the line thinning algorithm and line search procedure after the projected line image was segmented by figure 6.8 (c). After line correspondence and image rectification, matched points between the stereo pairs were found. The three-dimensional coordinate information was recovered by using the collinearity equations. Figure 6.8 (e) shows the three-dimensional shape of the test subject and the unwanted areas.

To reconstruct only the trunk surface, the boundary of the trunk in figure 6.8 (a) was manually outlined. Figure 6.9 (a) shows an image after the patient trunk was

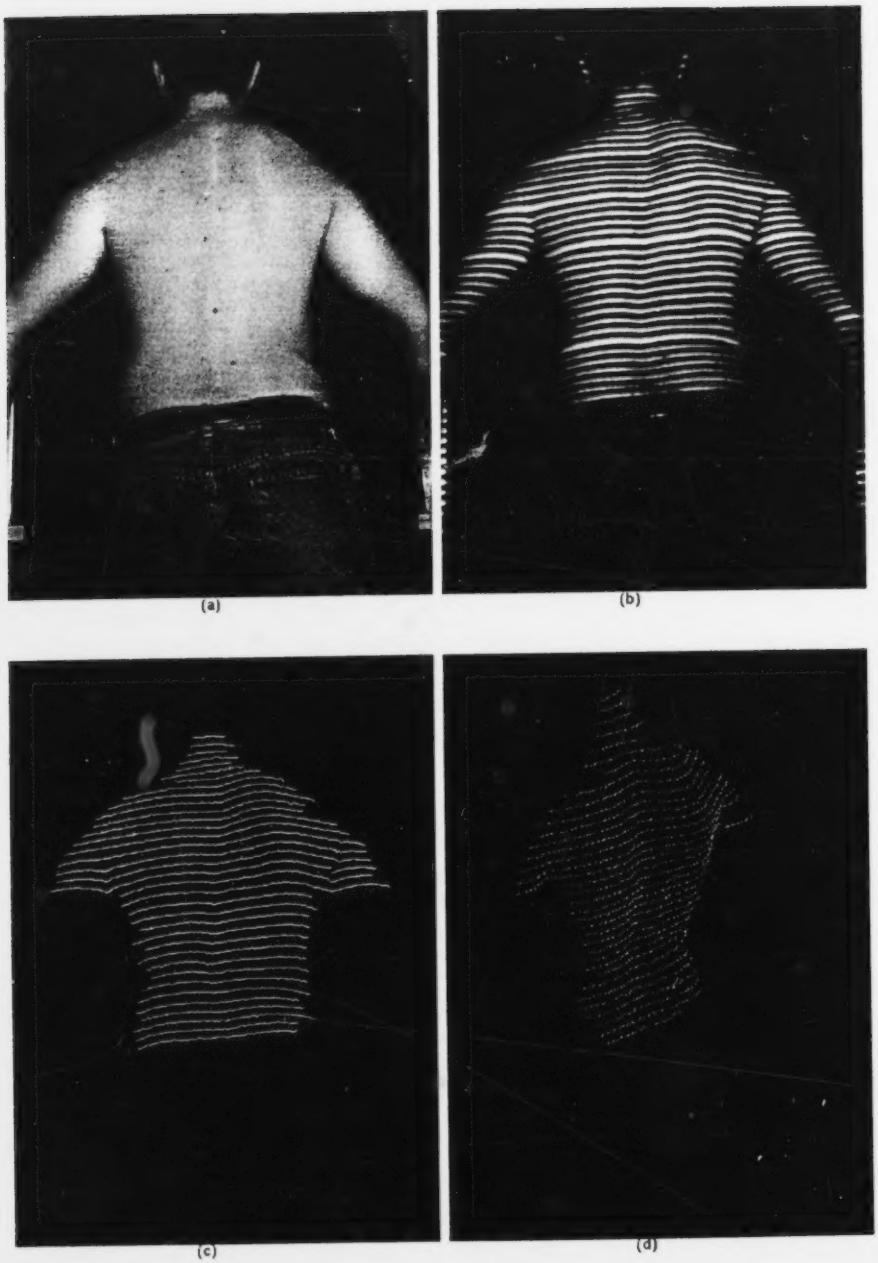

Figure 6.7: Sequence of processing steps for the scoliosis patient.

manually outlined by xpaint [36]. Figure 6.9 (b) shows the resultant edge image. Figure 6.9 (c) shows the results of the line thinning algorithm and line search procedure after the projected line image (figure 6.8 (b)) was segmented. Figure 6.9 (d) shows the reconstructed trunk surface.

The tests performed on the scoliosis patients show the feasibility of the image acquisition system in reconstructing human trunk surfaces. The test performed on the female subject shows when the intensity of the unwanted background information is similar to the intensity of the trunk, the system can not define the boundary of the trunk correctly. A possible solution is when imaging female subjects using a dark colour gown to enhance the contrast of trunk region. The system is able to reconstruct the surface of the female trunk after manually enhancing the contrast between the patient and the background.

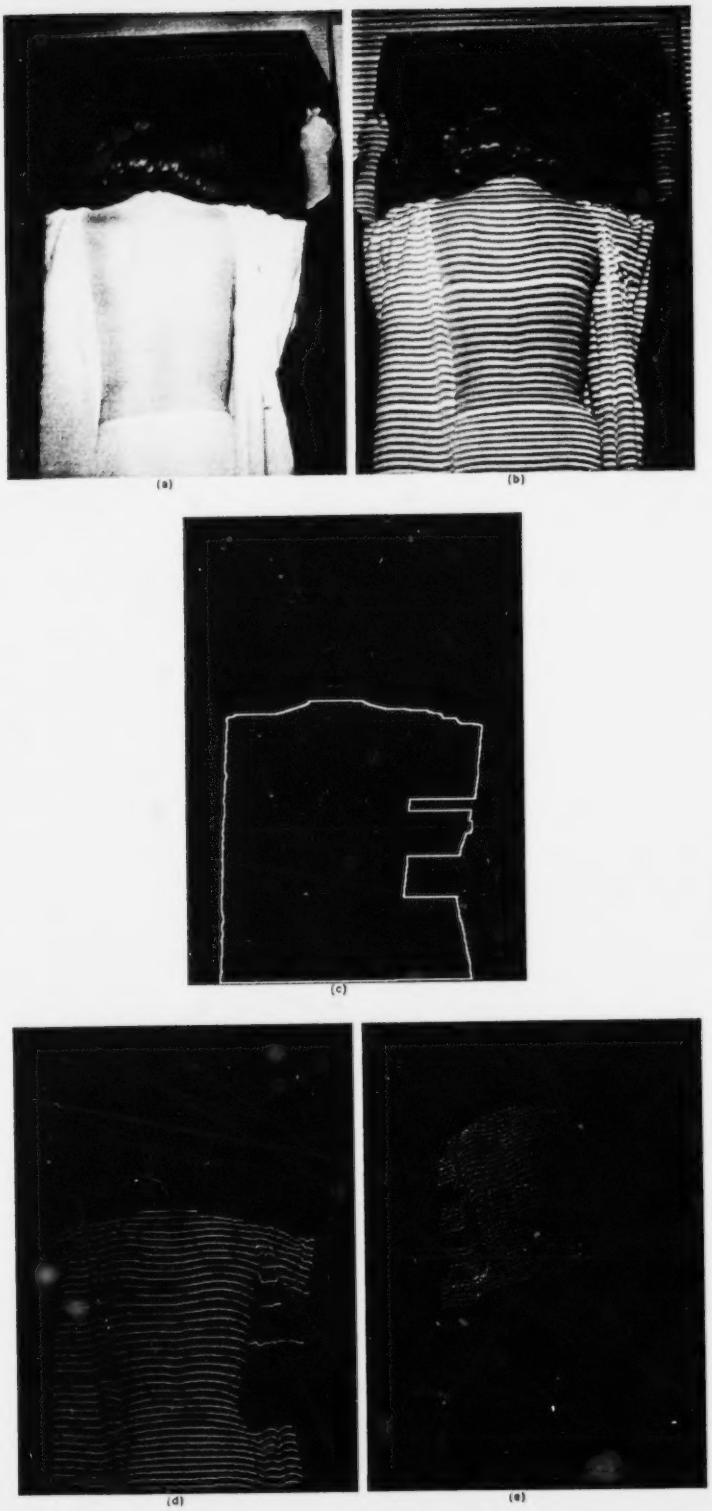

Figure 6.8: Sequence of processing steps for the scoliosis patient.

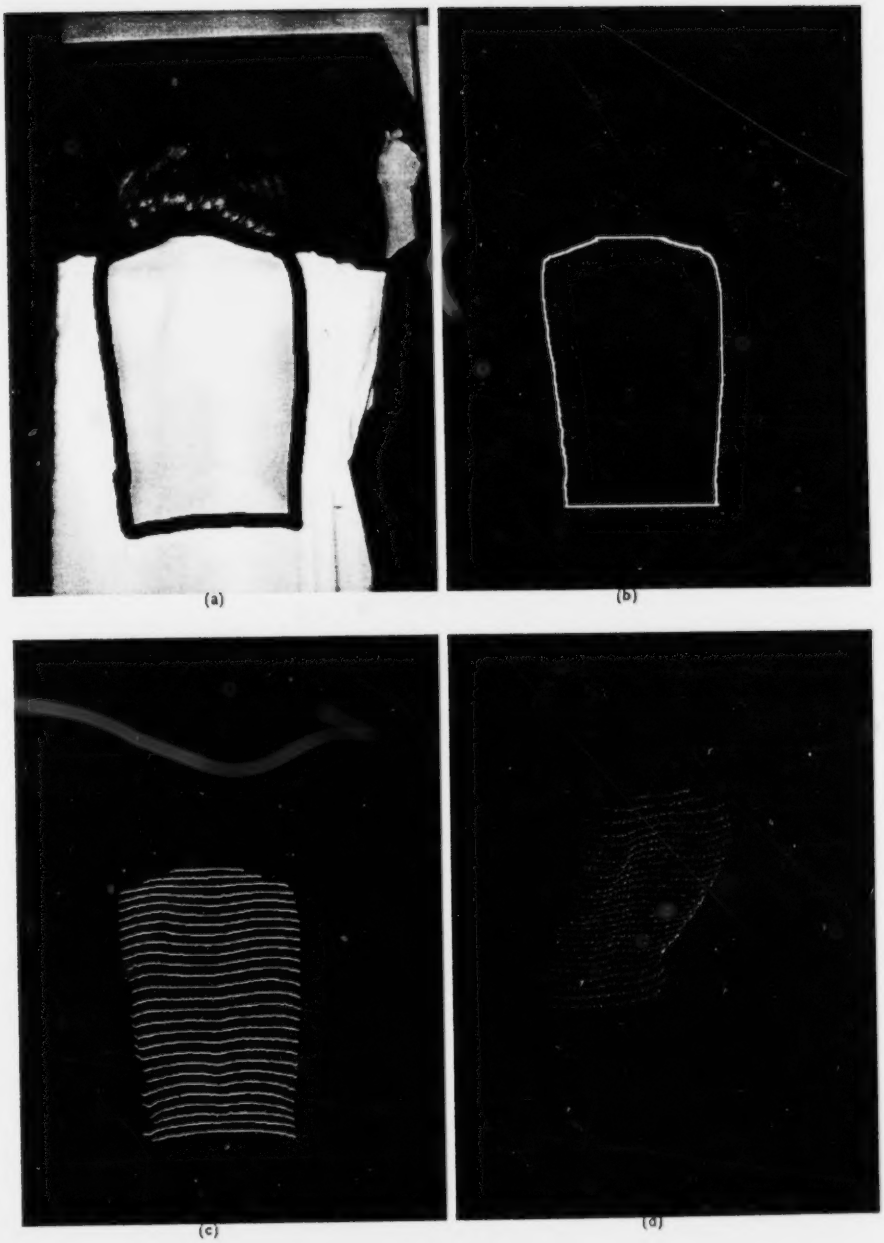

Figure 6.9: Sequence of processing steps for the scoliosis patient.

Chapter 7

CONCLUSION

7.1 Conclusion

The objective of this research was to develop a method to acquire three-dimensional coordinate information of the trunk surfaces with minimum human intervention, and to provide clinicians with a tool to assist in the evaluation of scoliotic deformities.

A non-invasive three-dimensional data acquisition system has been described. It is based on stereographic principles and requires only standard camera calibration and reconstruction procedures. This system is capable of recovering three-dimensional coordinate information of a human trunk. In this system, the three-dimensional data acquisition process consisted of image data acquisition, pattern recognition, image rectification, and three-dimensional reconstruction. A pair of projected line images and a pair of object images illuminated by a spot light were used to recover the three-dimensional data of the trunk surface. These images were captured by using the image acquisition system described in chapter 3. Region growing, edge detection, image thresholding, region thinning, a line linking procedure, eight connectivity and line drawing were used to define the one-pixel-thick lines. The correspondence lines between the stereo pair were found based on the relative thickness and brightness of the lines and the order of the lines. After image rectification, the matched points

between two images were found by searching points that were on the same scan line along the matched lines. The collinearity equations were used in recovery of the three-dimensional information.

Tests using solid models and scoliosis patients were conducted. Based on the experimental results from test models, the image processing methods and algorithms used have been shown to be able in recovering accurate and reliable three-dimensional data. The tests done on the scoliosis patients demonstrated the feasibility of the system to recovery three-dimensional information of the trunk surfaces.

This system will later be used in clinical trials to monitor the progress of scoliotic deformities. With this system, the clinician can recover the three-dimensional information of the trunk surfaces of the scoliosis patients. Combined with the graphical display of a computer three-dimensional surface model available at the Gelnrose Rehabilitation Hospital, the package provides clinicians with a tool to assist in the evaluation of scoliotic deformities.

The software implemented for the system consists of ten programs. This 3D reconstruction package requires the user to input a pair of projected line images, a pair of object images illuminated by a spot light, and a file contained with exterior and interior parameters of the cameras. The package automatically outputs X, Y and Z coordinates of the object surfaces in a file. Human intervention is not required. A detailed description of the 3D reconstruction package is given in Appendix 1.

7.2 Limitations

The system is capable of recovering the three-dimensional shape of the trunk surfaces. However, when there are sudden change of depth in the surfaces, lines projected onto the surfaces are occluded and/or falsely interconnected. The system does not have the

necessary constraints and knowledge to solve this problem. It will proceed, extracting one-pixel-thick lines, determining lines between the two images and, computing X, Y and Z coordinates for the surfaces. As a result, invalid information may be generated.

The system also has limitations in measuring three-dimensional coordinates of small object surfaces. This is because the computation of three-dimensional data requires thick lines to be corresponded first. If the thick lines are missed from the projected line images, line correspondence between two images can not proceed. The system will output a warning message.

A pair of object images illuminated by a spot light is needed to define the region of interest for the projected line images. If there is movement of the patient, an incorrect boundary will be generated. This will cause the system to produce extra data points that are off the body or to lose information from the trunk surfaces. However, because the four images can be captured within 1-2 seconds by the two cameras, involuntary movement of the patient due to breath and posture sway is likely to be small. This small amount of movement will generally not affect performance of the system.

The matching algorithm used in the system is based on the connectivity of the lines. If the broken line segments are not connected into the continuous lines, portions of the line segments may be omitted from the results. On the other hand, if the broken line segments are all connected into the continuous lines using linear interpolation, inaccurate data may be yielded in the results. In this case, it is a trade off between the completeness and accuracy of the data.

If the coordinates of an arbitrary object point P projected onto the two image planes are (x_1, y_1) and (x_2, y_2), the approximate Z coordinate of the object point can be calculated based on the following equations.

$$Z = \lambda f_1 + Z_{c1} \tag{7.1}$$

where the λ is:

$$\lambda = \frac{y_2(X_{c2} - X_{c1}) - x_2(Y_{c2} - Y_{c1})}{x_1 y_2 - x_2 y_1}. \tag{7.2}$$

Where (X_{c1}, Y_{c1}, Z_{c1}) and $(X_{c2}, Y_{c2}$, and $Z_{c2})$ are the coordinates of the perspective centers and f_1 and f_2 are the focal length of both cameras.

When the coordinates of the corresponding arbitrary points between the top and the bottom camera images were (261, 228) and (268, 182), by using the Eqs. 7.1 and 7.2, the Z coordinate of the object point was found to be 178 mm. If 6.5mm and 1.5mm were added to the image coordinates x and y, respectively for both images, the Z coordinate of the object point was then 176 mm. As a result, the change in Z coordinate (ΔZ) was 2 mm when 6.5mm and 1.5mm were added to the image coordinates x and y, respectively. By increasing the density of the lines projected onto the surface, the change in Z coordinate between consecutive lines will become smaller. This would increase the resolution of the system.

7.3 Future Work

The density of the three-dimensional data in the vertical direction can be improved by re-processing the negative of the projected line images. The thinned lines from the dark band between the lines can be superimposed onto the present thinned line image. As a result, the density of data in Y coordinate (the vertical direction) is increased.

The detection of the boundary for the region of interest can be improved. At present, a black curtain is used in the background to increase the contrast of the image. It is possible that by using a better boundary detection technique, and the black curtain can be removed. Thus, the efficiency of the system is increased.

The system is able to reconstruct an accurate three-dimensional cylindrical surfaces, triangular shaped object surfaces and step pattern object surfaces. Moreover, it is capable of recovering the three-dimensional information of the human subjects with mild scoliosis. Further investigations on the accuracy of the reconstructed human trunk surfaces are required.

Bibliography

[1] Y.I. Abdel-Aziz and H.W Karara. Direct linear transformation from comparator coordinates into object space coordinates. *ASP Symposium on Close-Range Photogrammetry*, 1971.

[2] M. Assoul, M. Zahidi, P. Corcuff, and J. Mignot. Three-dimensional measurements of skin surface topography by triangulation with a new laser profilometer. *Journal of medical Engineering and Technology*, 18(1):11–21, 1994.

[3] N. Ayache. *Artificial Vision for Mobile Robots.* The MIT Press, London, England, 1991.

[4] N. Ayache and C. Hansen. Rectification of images for binocular and trinocular stereovision. *Proc. International Conference on Pattern Recognition*, 9:11–16, 1988.

[5] H.H. Baker and T.O. Binford. Depth from edge and intensity based stereo. *Proceedings of 7th International Joint Conference on Artificial Intelligence*, pages 631–636, 1981.

[6] J. Bardley. xv - version 3.00, 1993. Copyright 1993, John Bardley.

[7] J.R. Beveridge. Segmenting images using localized histograms and region merging. *International Journal of Computer Vision*, 2:311–347, 1989.

[8] H.A. Beyer. Some aspects of the geometric calibration of ccd-cameras. *Fast Processing of Photogrammetric Data*, pages 68–81, 1987.

[9] W.P. Bunnel. An objective criterion for scoliosis screening. *Journal of Bone and Joint Surgery*, 66-A:1382–1387, 1984.

[10] R.G. Burwell. Standard trunk symmetry scores. *Journal of Bone and Joint Surgery*, 65-B:452–460, 1983.

[11] W. Cho. Transformation of the pixel system to the vertical position and resampling. Technical Notes in Photogrammetry 3, Ohio State University, Dept. of Geodetic Science and Surveying, Columbus, Ohio, 1989.

[12] S.D. Cochran and G. Medioni. 3d surface description from binocular stereo. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 14(10):981–987, 1992.

[13] J.L. Crowley, P. Bobet, K. Sarachik, S. Mely, and M. Kurek. Mobile robot perception using vertical line stereo. *Robotics and Autoaomous Systems*, 7:125–138, 1991.

[14] B. Drerup. 3d acquisition, reconstruction and modelling techniques applied on scoliotic deformity. *Symposium on 3D Scoliotic Deformity*, pages 2–10, 1992.

[15] S.M. Dunn, R.L. Keizer, and J. Yu. Measuring the area and volume of the human body with structured light. *IEEE Trans. on Systems, Man, and Cybernetics*, 19(6):1350–1364, 1989.

[16] D. Elad and S. Einav. Three-dimensional measurement of biological surfaces. *ISPRS. Journal of Photogrammetry and Remote Sensing*, 45:247–266, 1990.

[17] O. Faugeras. *Three-Dimensional Computer Vision.* The MIT Press, London, England, 1993.

[18] R.B. Fisher. *From Surfaces to Objects.* John Wiley and Sons, New York, NY, USA, University of Edinburgh, Edinburgh, UK, 1988.

[19] J.D. Foley, A.V. Dam, S.K. Feiner, and J.F. Hughes. *Computer Graphics Principles and Practice.* 2 edition, 1990.

[20] W. Forstner. A feature based correspondence algorithm for image matching. *International Archives of Photogrammetry*, 26-III, 1986.

[21] W Frobin and E Hierholzer. Automatic mesurement of body surface using rasterstereography. *Photogrammetric Engineering and Remote Sensing*, 49(3):377–384, 1983.

[22] W Frobin and E Hierholzer. Calibration and model reconstruction in analytical close-range stereophotogrammetry. *Photogrammetric Engineering and Remote Sensing*, 48(1):67–72, 1988.

[23] W. Frobin and E. Hierhozer. Video rasterstereography: A method for on-line measurement of body surface. *Photogrammetric Engineering and Remote Sensing*, 57(10):1341–1345, 1991.

[24] J.G Fryer and S.O. Mason. Rapid lens calibration of a video camera. *Photogrammetry Engineering and Remote Sensing*, 55(4):437–442, 1989.

[25] R.C. Gonzalez and R.E. Woods. *Digital Image Processing.* Addion-Wesley, New York, 1993.

[26] A. Gregory and R.T. Lipczynski. The three dimensional reconstruction of human facial images. *Proceedings of the 15th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, I-130:50–53, 1993.

[27] W.E.L. Grimson. Computational experiments with a feature based stereo algorithm. *IEEE Transactions On Pattern Analysis and Machine Intelligence*, PAMI-7(1):17–34, 1985.

[28] M.J. Hannah. A system for digital stereo image matching. *Photogrammetric Engineering and Remote Sensing*, 55(12):1765–1770, 1989.

[29] R. Hartle and R. Gupta. Computing matched-epipolar projections. *IEEE Computer Society Conference on Computer Vision and Pattern Recognization*, pages 549–555, 1993.

[30] E. Hierholzer. Improved methods of image data processing in video rasterstereography. *Surface Topography and Spinal Deformity*, VI:29–33, 1992.

[31] D. Hill, V.J. Raso, N.G. Durdle, and A.E. Peterson. Designing a video based technique for trunk measurement. *Int. Symposium on 3D Scoliotic Deformity*, pages 157–161, 1992.

[32] D.Q. Huynh and R.A. Owens. Line labelling and region segmentation in stereo image pairs. TR TR 92/7, University of Western Australia, Dept. of Computer Science, Nedlands, Australia, 1992.

[33] R.A. Jarvis. A perspective on range finding techniques for computer vision. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, PAMI-5(2):122–139, 1983.

[34] H.M. Karara. *Non-Topographic Photogrammetry*. American Society for Photogrammetry and Remote Sensing., 2nd edition, 1989.

[35] M. Keefe. Capturing facial surface information. *Photogrammetric Engineering and Remote Sensing*, 52(9):1539–1548, 1986.

[36] D. Koblas. xpaint - version 2.1, 1993. Copyright 1993, David Koblas.

[37] J.R.T Lewis and T. Sopwith. Measuring the human chest with structured lighting. *Pattern Recognition Letter*, 4:359–366, 1986.

[38] J.R.T. Lewis and T. Sopwith. Three dimensional surface measurement by microcomputer. TR 1, IBM UK Scientific Centre, Winchester, UK, 1986.

[39] M. Li. Hierarchical multipoint matching. *Photogrammetric Engineering and Remote Sensing*, 57(8):1039–1047, 1991.

[40] S.A. Lloyd. Binocular stereo algorithm based on the disparity-gradient limit and using optimization theory. *Image and Vision Computing*, 3:177–181, 1985.

[41] D. Marr and T. Poggio. A theory of human stereopsis. *Proceedings of The Royal Society*, 204:301–328, 1979.

[42] J.E.W Mayhew and J.P. Frisby. *3D Model Recognition From Stereoscopic Cues.* The MIT Press, London, England, 1991.

[43] F.H. Moffitt and E.M. Mikhail. *Photogrammetry.* Harper and Row, University of California, Berkeley and Purdue University, 3rd edition, 1980.

[44] V.S. Nalwa. *A Guided Tour of Computer Vision.* Addison-Wesley Publishing Company, AT&T Bell Laboratories, 1993.

[45] J.D. Pearson, P.H. Dangerfield, J.T. Atkinson, J.B. Gomm, J.C. Dorgan, C.A. Hobson, and D.H. Harvey. Measurement of boby surface topography using an automated imaging system. *Acta Orthopaedic Belgica*, 58:73–79, 1992.

[46] J.D. Pearson, P.H. Dangerfield, C.A. Hobson, and Y. Li. An automated visual system for the measurement of the three-dimensional deformity of scoliosis. *Surface Topography and Spinal Deformity*, VI:50–56, 1992.

[47] A.E. Peterson, N.G. Durdle, V.J. Raso, and D.L. Hill. Calibration of video cameras for scoliosis mapping. *Geomatica*, 47:29–38, 1993.

[48] P.K. Sahoo, S. Soltani, and A.K.C. Wong. A survey of thresholding techniques. *Computer Vision, Graphics, and Image Processing*, 41:233–260, 1988.

[49] Y. Shirai. *Three-Dimensional Computer Vision.* Springer-verlag, New York, 1986.

[50] J.C. Simon. *From Pixels To Features.* Elsevier Science Publishers B.V., Paris, France, 1989.

[51] H.P. Trivedi and S.A. Lloyd. The role of disparity gradient in stereo vision. *Perception*, 14:685–690, 1985.

[52] A.R. Turner-Smith. A television/computer three-dimensional surface shape measurement system. *J. Biomechanics*, 21:515–529, 1988.

[53] A.R. Turner-Smith, J.D. Harris, G.R. Houghton, and R.J. Jefferson. A method for analysis of back shape in scoliosis. *Journal of Biomechanics*, 21(6):497–509, 1988.

[54] A.R. Turner-Smith and D.C. Thomas. Some relationships between the spine and skin surface shape. *Surface Topography and Spinal Deformity*, IV:3–11, 1988.

[55] R.G. White and D.A. Perednia. Automatic derivation of initial match points for paired digital images of skin. *Computerized Medical Imaging and Graphics*, 16(3):217–225, 1992.

[56] P.R. Wolf. *Elements of Photogrammetry.* McGraw-Hill Inc., New York, U.S.A., 2 edition, 1983.

Appendix 1

Description of The 3D Reconstruction Package

In this appendix the description of the 3D reconstruction package implemented for the three-dimensional data acquisition system is given. It gives an idea about the actual implementation of the package and is intended to supplement the information already available in the preceding chapters.

The current version of the 3D reconstruction package is implemented in the C programming language and runs on IBM RISC System/6000 workstations. It requires the user to input a pair of projected line images, a pair of object images illuminated by a spot light, and a file contained with exterior and interior parameters of the cameras. This package automatically outputs X, Y and Z coordinates of the object surfaces in a file. Human intervention is not required.

The 3D reconstruction package reads and writes Grey scale tiff image. Tiff image I/O is done via the libtiff.a library. This library is located in /usr/local/lib. Header files of the Tiff images can be found in /usr/local/include library and are linked to the default lib and include directories. The Tiff images from Mac. do not have the same header as the Tiff images used IBM RISC System/6000 workstation. It is required to save the Mac's Tiff images as the Tiff images in the IBM RISC System/6000 workstation using xv [6] before using them in the package.

This appendix is divided into two sections to reflect the organization of the preceding chapters. The descriptions of the implementation of region growing, edge detection, image thresholding, region thinning, line linking procedure, eight connectivity and line drawing are given in section one. Section two describes the implementation of line correspondence between the stereo image pair, image rectification, point correlation and three-dimensional recovery.

1.1 Implementation of Image processing techniques

Procedure region_grow() is used to define the region of interest. It groups pixels that have similar intensity values into regions which define the region the interest. It starts at a seed point in the object, and moves toward the boundaries of the object in each of the partitioned regions, and includes all candidate points that have similar properties. This procedure is called first in the package. The declaration of the region_grow() procedure is:

```
void region_grow(image,Seed)
ImagePt image;
cor_type *Seed;
```

The parameter "image" is the address of(a pointer to) a data structure defined by ImagePt. The parameter "Seed" is the address of a data structure defined by cor_type. The ImagePt data structure is declared as:

```
typedef struct{
  int W;
  int H;
  int max;
  int **data;
} Image_Type;
```

```
typedef Image_Type *ImagePt;
```

The ImagePt data structure is used to store the image information. The W and H fields of this structure contain the width and height of image, respectively. The max field is the maximum intensity value of the image. The data is a pointer to a two-dimensional array contained with each pixel intensity values of the image. The input image is a object image illuminated by a spot light. The procedure returns a image of region of interest.

The cor_type data structure is used to store the coordinates of the seed point found in the region_grow procedure. Its declaration is:

```
typedef struct {
  int imx, imy;
} cor_type;
```

Procedure boundary_detection() is used to define the boundary of the region of interest. The Sobel operator masks are used to detect the edge of the region of interest. The Sobel operators calculate the gradient of the pixels in the image. The edges in the image are found using the magnitude of the pixel gradient values. If the magnitude of the pixel gradient value is greater than 40 percent of the difference between the minimum and maximum intensity values in the image, this pixel is an edge pixel. The declaration of the boundary_detection() procedure is:

```
void boundary_detection(image)
ImagePt image;
```

The procedure inputs the image generated from the region_grow procedure. After noise reduction and edge linking, it returns a complete boundary of the region of interest.

The local_threshold() procedure is used to enhance the contrast of the projected lines. The projected line image is partitioned into sub-images and a threshold value is determined for each of the sub-images. A 10 by 10 pixels window for local thresholding is used. The mean intensity value of each sub-image is used to separate the object (the projected lines) from background (the object). Image thresholding is accomplished by scanning the image pixel by pixel and labelling each pixel as object or background, depending on whether the intensity value of that pixel is greater or less than the mean intensity value. The declaration of the local_threshold() procedure is:

```
void local_threshold(image)
ImagePt image;
```

The procedure inputs the projected line image and outputs an enhanced projected line image.

Procedure line_thinning() is used to determine the one-pixel-thick lines from the projected lines in the images. The skeletonizing method (line thinning algorithm) is used to thin the projected lines. Line thinning shrinks each projected line by repeatedly deleting edge points. The thinned lines are approximately at the center of the original projected lines. The algorithm operates under three basic constraints: 1) it does not remove end points. 2) it does not break the connectedness of the line, and 3) it does not cause excessive erosion of the region. The thinning algorithm applies a two-step procedure described in section 4.2 iteratively until no further edge points are deleted at which time the algorithm terminates and the resulting lines are approximately at the center of the original projected lines. The declaration of the line_thinning() procedure is:

```
void line_thinning(image)
ImagePt image;
```

The procedure inputs the thresholded projected lines and outputs the thinned lines.

Procedure eight_connectivity() is used to identify and number the continuous lines in the image. Eight-connectivity classified each line(connected components) only once. The image is scanned pixel by pixel, from left to right and from top to bottom to label and identify the lines in the image. At the end of the scanning process, all points with value 1 are labelled, but some of the labels are equivalent. The equivalent label pairs are sorted into equivalence classes, and a unique label is assigned to each class. Then a second pass through the image replaces the equivalent labels by the label assigned to the class. At the end of the eight-connectivity procedure, each line in the image has a unique label assign to it. The declaration of the eight_connectivity() procedure is:

```
void eight_connectivity(image,G_Classes,G_Num)
ImagePt image;
LineSet_Type **G_Classes;
int *G_Num;
```

The parameter "G_Num" is used to store the numbers of the projected lines found in the image. The parameter "G_Classes" is the address of(a pointer to) a data structure defined by LineSet_Type. It is used to store each connected region. The LineSet_Type data structure is declared as:

```
typedef struct {
cor_type *cor;
   int GP;
   int GPSize;
   int match_flag;
} LineSet_Type;
```

The LineSet_Type data structure is used to store the projected line information. The GP and GPSize fields of the structure contain the labels and the pixel numbers

of a projected line(groups of pixels), respectively. The match_flag field is used to indicate the status of line correspondence. The cor is a pointer to a one-dimensional array containing the coordinates of each pixel of a line. The procedure inputs the thinned line image and outputs a set of connected lines and the number of the lines.

Procedure ConSeg() is used to connect the broken line segments to produce continuous lines. It is based on the smoothness criterion, and it reconnects broken line segments based on the distance and orientation of the line segments. The procedure starts with finding both start and end points of the lines in the line set. the distances between a start point of a line and the end points of the remaining lines are calculated. If the distance is smaller than a threshold value of 10 pixels, these two line segments are considered as a candidate pair of broken lines that potentially might be reconnected. The inclination angle between the start and end points of the candidate lines is examinated. If the inclination angle is less than ± 45 degrees, these two line segments are joined. This procedure is applied repeatly, stopping when no further broken line segments are found. The declaration of the ConSeg() procedure is:

```
void ConSeg(image, classes, num_classes)
ImagePt image;
LineSet_Type *classes;
int num_classes;
```

The procedure inputs the line set found in the eight_connectivity procedure and outputs a connected line set. The parameter "num_classes" is used to store the number of the connected lines.

Procedure spatial_relation() is used to connect a series of neighbouring broken lines to continuous lines. It is based on the constraints that the projected lines can never cross each other and are contiguous. A two-step procedure is implemented to connect these lines. The procedure starts with searching the candidates of the

broken lines that potentially may be reconnected. Broken lines are candidates for reconnection if the distance between the start point of a line and the end points of the rest of the lines is less than 30 pixels. Based on the constraints that projected lines are contiguous and never cross each other, the candidate segments are examinated. If the constraints described in section 4.3 are satisfied, the candidate segments are connected. The declaration of the spatial_relation() procedure is:

```
void spatial_relation(image,classes,num_classes)
ImagePt image;
LineSet_Type *classes;
int num_classes;
```

The procedure inputs the line set after the ConSeg() procedure and outputs a updated connected line set.

Procedure draw_line() is used to connect the disruption of the lines. This procedure computes the coordinates of the pixels that lie on or near a straight line defined by the end points. It connects the broken line segments with an interpolation of a straight line. The procedure consists of three steps: First, computes the initial decision variable d_{start} described in section 4.3. Second, chooses the next pixel based on the sign of the decision variable calculated in the previous step. Third, increments the decision variable by adding either deltaE or deltaNE described in section 4.3 to the current value. The next pixel is selected based on the sign of the new decision variable. This process is repeated until the broken lines are joined at which time the algorithm terminates. The declaration of the draw_line() procedure is:

```
void draw_line(x1,y1,x2,y2,k,classes)
int x1,y1;
int x2,y2;
int k;
```

```
LineSet_Type *classes;
```

The procedure inputs both end points of the line (x_1, y_1) and (x_2, y_2). and outputs a line drawn by the procedure in a pointer array "classes". "k" is the number of the points in the drawn line.

1.2 Implementation of Three-Dimensional Reconstruction

Procedure FindThickLine() is used to identify the thick lines in the projected line images. The thick lines are identified based on the relative thickness and brightness. There are three steps to find the thick lines in the images. First, the total and the mean intensity value of the cross section of the projected lines are calculated. The projected line with the maximum total intensity value is selected and considered to be one of the thick lines. Second, after one of the thick lines is identified, the absolute differences in mean intensity value between the selected thick line and the remaining projected lines are calculated. If the difference is less than a threshold value of 10, the lines are considered to be thick line candidates. In the third step, a precoded line pattern relationship between the thin and the thick lines described in section 3.2 is used. If the number of the thin lines between the selected thick line and the thick line candidates satisfy the precoded line pattern constraint, the candidate lines are considered to be thick lines. The declaration of the FindThickLine() procedure is:

```
void FindThickLine(image, thickline,thick_num,seed)
ImagePt image;
Thick_type *thickline;
int *thick_num;
cor_type *seed;
```

The parameter "image" stores the input of the projected line image. The parameter "thickline" is a pointer to a data structure defined by Thick_type. It is used to

store the thick lines found in the procedure. The parameter "thick_num" stores the number of the thick lines found. "seed" contains the coordinates of the start point of the search procedure. The Thick_type data structure is declared as:

```
typedef struct {
  int mean, label;
  int Xmin, Xmax;
  int Ymin, Ymax;
  int MatchFlag;
} Thick_type;
```

The Thick_type data structure is used to store the thick line information. The mean and label fields of this structure contain the mean intensity value and the labels of the thick line, respectively. The coordinates of the end points of the thick line are stored in (Xmin, Ymin) and (Xmax, Ymax). The MatchFlag field is used to indicate the status of line correspondence.

The thick lines are corresponded based on the relative image y coordinate of the thick lines. A thick line with the maximum y coordinate in one image matches with a thick line with the maximum y coordinate in another image. A thick line with the second largest y coordinate in one image matches a thick line with the second largest y in another image. This matching procedure is continuous until all the thick lines in both images are correlated. The declaration of the MatchThickLine() procedure is:

```
void MatchThickLine(col,label1,label2,
set1,set2,set1_num,set2_num,match)
MatchLine_Type *match;
LineSet_Type *set1,*set2;
int label1,label2, *col;
```

The parameter "match" is a pointer to a data structure array defined by MatchLine_Type. It is used to store the matched thick lines. The parameter "set1" and "set2" contain the projected lines of the top and bottom camera images, respectively. The parameter "label1" and "label2" contain the line label of the top and bottom camera images. The MatchLine_Type data structure is declared as:

```
typedef struct {
  SetIndex_Type *Index;
  int NumOfMatchLine;
} MatchLine_Type;
```

The MatchLine_Type data structure is used to store the matched thick lines found in the MatchThickLine() procedure. The NumOfMatchLine fields of this structure contain the number of the matched thick lines between two images. The Index field is a pointed to a data structure defined by SetIndex_Type. The declaration of the SetIndex_Type data structure is:

```
typedef struct {
  int LSet1;
  int LSet2;
} SetIndex_Type;
```

The LSet1 and LSet2 fields of the SetIndex_Type structure contain the index number of the line set of the top and bottom camera images, respectively.

Procedure Match_Line() is used to match the corresponding thin lines between two images. This procedure corresponds the thin lines by using the initial thick line matches. Line order and relative location of the thin and thick lines are used. It starts at one of the thick line matches and searches upward and downward in both images for the corresponding thin lines. If the order of two lines is the same in both images during the search procedure, these two lines are considered to be matched. The search

process terminates when the difference of distance between three consecutive lines in either image is greater than a threshold value of 5 pixels, or when all the projected lines are matched. The declaration of the Match_Line() procedure is:

```
void Match_Line(image1,image2,thickline1,
thickline2,Match,thick1_num,thick2_num,
Set1,Set2,set1_num,set2_num)
Thick_type *thickline1, *thickline2;
MatchLine_Type *Match;
int thick1_num, thick2_num;
ImagePt image1,image2;
LineSet_Type *Set1, *Set2;
int set1_num, set2_num;
```

The parameters "thickline1" and "thickline2" are pointers to a data structure array defined by Thick_type. They contain the initial matched thick lines in the top and bottom camera images, respectively. The parameter "Match" is a pointer to a array containing the corresponding projected lines between the top and bottom camera images. The parameters "Set1" and "Set2" are pointers to structure data arrays containing the projected lines in the top and bottom camera images, respectively. The parameter "thick_num" and "set_num" contain the number of the thick lines and the number of the projected lines in the images, respectively.

Procedure RectImage() is used to make an equivalent vertical image pair from a tilted image pair. The resulting image pair is recorded in epipolar geometry. The scan lines of stereo pairs are epipolar lines parallel to the axes of the image coordinate frames. The first stage of this procedure is to compute the orthogonal rotation matrix which transforms the pixel image pair into the vertical image pair. The second stage involves a transformation from the vertical image to the normalized rectified image. The declaration of the RectImage() procedure is:

```
void RectImage(image,Rn,CD,XValue,YValue,cent)
ImagePt image;
double Rn[3][3];
Rectify_Type CD;
int **XValue, **YValue;
cor_type *cent;
```

The parameter "Rn" is a 3 by 3 array containing the elements of the orthogonal rotation matrix. The parameter "CD" is defined by a data structure Rectify_Type. It contains the exterior and interior parameters of the camera. The declaration of the data structure Rectify_Type is:

```
typedef struct {
  double omega;
  double phi;
  double kappa;
  double zp;
  double cx;
  double cy;
  double cz;
  double k1;
} Rectify_Type;
```

The omega, phi and kappa fields of this data structure contain the rotation angles of the camera. The cx, cy and cz store the spatial world coordinates of the perspective center of the camera. The zp and k1 contain the focal length and radial correction of the camera, respectively.

Procedure Match_YCor is used to match the corresponding points between the top and bottom camera images. Corresponding points are found by searching points

that are on the same scan line along the matched lines in two images, The declaration of the Match_YCor() procedure is:

```
void Match_YCor(MatchLines,LineSet1,LineSet2,
MatchPoints,RectLineSet1,RectLineSet2)
MatchPoint_Type *MatchPoints;
LineSet_Type *LineSet1, *LineSet2;
LineSet_Type *RectLineSet1, *RectLineSet2;
MatchLine_Type *MatchLines;
```

The parameter "MatchPoints" is a pointer to a data structure array defined by MatchPoint_Type. It contains the matched points found between the two images. The parameter "RectLineSet" and "LineSet" are pointers to the data structure array containing the rectified projected lines and the projected lines of the images, respectively. The parameter "MatchLines" is a pointer to the array storing the matched lines. The declaration of the MatchPoint_Type data structure is:

```
typedef struct {
  cor_type *PointSet1,*PointSet2;
  int NumOfMatchP;
} MatchPoint_Type;
```

Procedure CalcXYZ() is used to compute the X, Y and Z coordinate information of the object surfaces. The collinearity equations described in section 2.5 are used. The equations are based on the assumption that the object point, the perspective center of the camera, and the image point lie on a straight line. The equations using the least squares solution compute the X, Y, and Z coordinates by an iterative process, stopping when the corrections become less than 0.0000001. The declaration of the CalcXYZ procedure is:

```
void CalcXYZ (MatchPoints,CamData,NumLine)
MatchPoint_Type *MatchPoints;
CamData_Type CamData;
int NumLine;
```

The parameter "MatchPoints" is a pointer to the data structure array containing the matched points. The parameter "CamData" is defined a data structure CamData_Type. It contains the exterior and interior parameters of both cameras. The declaration of the data structure CamData_Type is:

```
typedef struct {
  double cx[2];
  double cy[2];
  double cz[2];
  double omega[2];
  double phi[2];
  double kappa[2];
  double f[2];
  double xo[2];
  double yo[2];
  double k1[2];
  double k2[2];
  double p1[2];
  double p2[2];
}CamData_Type;
```

The omega, phi and kappa fields of the data structure contain the rotation angles of both cameras. The cx, cy and cz store the world coordinates of the perspective centers of both cameras. The f, k1, k2 , p1 and p2 contain the focal length, radial and de-centering corrections of both cameras, respectively.